Junos® OS

Broadband Subscriber VLANs and Interfaces Feature Guide

Release

14.1

Modified: 2014-12-22

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

Junos® OS Broadband Subscriber VLANs and Interfaces Feature Guide
14.1

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

YEAR 2000 NOTICE

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

END USER LICENSE AGREEMENT

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

Table of Contents

List of Figures

List of Tables

Part 8 Configuration Statements and Operational Commands

About the Documentation

Documentation and Release Notes

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

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

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

Supported Platforms

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

- MX Series

Using the Examples in This Manual

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

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

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

Merging a Full Example

To merge a full example, follow these steps:

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

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

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

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

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

Merging a Snippet

To merge a snippet, follow these steps:

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

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

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

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

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

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

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

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

Documentation Conventions

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

Table 1: Notice Icons

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

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

Table 2: Text and Syntax Conventions

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

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

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

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

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

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

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

Documentation Feedback

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

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

Requesting Technical Support

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

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

Self-Help Online Tools and Resources

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

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

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

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

Opening a Case with JTAC

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

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

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

PART 1

Configuring Dynamic VLANs for Subscriber Access Networks

CHAPTER 1

Dynamic VLAN Overview

Subscriber Management VLAN Architecture Overview

The subscriber management logical network architecture is as important as the physical network architecture. You configure the logical portion of the subscriber management network using virtual local area networks (VLANs).

Subscriber management uses three VLAN models:

- **Service VLAN**—The service VLAN (S-VLAN) provides many-to-one (N:1) subscriber-to-service connectivity: The service VLAN carries a service (for example, data, video, or voice) to all subscribers instead of having different services share a VLAN. Adding a new service requires adding a new VLAN and allocating bandwidth to the new service. The service VLAN model enables different groups that are using the broadband network (for example, external application providers) to manage a given service. One limitation of service VLANs is the absence of any logical isolation between user sessions at the VLAN level. This lack of isolation requires that the multiservice access node (MSAN) and broadband network gateway (BNG) provide the necessary security filtering.
- **Customer VLAN**—The customer VLAN (C-VLAN) provides one-to-one (1:1) subscriber-to-service connectivity: One VLAN carries all traffic to each subscriber on the network. Having a single VLAN per subscriber simplifies operations by providing a 1:1 mapping of technology (VLANs) to subscribers. You can also understand what applications any subscriber is using at any given time. Because you use only one VLAN to carry traffic to each subscriber, this approach is not affected when adding new services. However, using a pure C-VLAN model consumes more bandwidth because a single television channel being viewed by multiple subscribers is carried across the network several times—once on each C-VLAN. This approach requires a more scalable, robust edge router that can support several thousand VLANs.
- **Hybrid C-VLAN**—The hybrid VLAN combines the best of both previous VLANs by using one VLAN per subscriber to carry unicast traffic and one shared multicast VLAN (M-VLAN) for carrying broadcast (multicast) television traffic. You can use both the

pure and *hybrid* C-VLAN models in different portions of the network, depending upon available bandwidth and MSAN capabilities.

NOTE: The term *C-VLAN*, when used casually, often refers to a *hybrid* C-VLAN implementation.

We recommend using one of the C-VLAN models to simplify configuration and management when expanding services. However, some MSANs are limited to the number of VLANs they can support, limiting the ability to use either C-VLAN model.

NOTE: Most MSANs can support the service VLAN model.

Broadband Subscriber Management VLANs Across an MSAN

You configure VLANs to operate between the MSAN and the edge router (broadband services router or video services router). However, the MSAN might modify VLAN identifiers before forwarding information to the subscriber in the following ways:

NOTE: Not all MSANs support these options.

- The VLAN identifiers can be carried within the ATM VCs or they can be removed. The value of keeping the VLAN header is that it carries the IEEE 802.1p Ethernet priority bits. These priority bits can be added to upstream traffic by the residential gateway, allowing the DSLAM to easily identify and prioritize more important traffic (for example, control and VoIP traffic). Typically, a VLAN identifier of zero (0) is used for this purpose.
- In a C-VLAN model, the MSAN might modify the VLAN identifier so that the same VLAN is sent to each subscriber. This enables the use of the same digital subscriber line (DSL) modem and residential gateway configuration for all subscribers without the need to define a different VLAN for each device.

Customer VLANs and Ethernet Aggregation

The 12-bit VLAN identifier (VLAN ID) can support up to 4095 subscribers. When using an aggregation switch with a C-VLAN topology, and fewer than 4095 subscribers are connected to a single edge router port, the aggregation switch can transparently pass all VLANs. However, if the VLAN can exceed 4095 subscribers per broadband services router port, you must use VLAN stacking (IEEE 802.1ad, also known as Q-in-Q). VLAN stacking includes two VLAN tags—an outer tag to identify the destination MSAN and an inner tag to identify the subscriber. For downstream traffic (that is, from the broadband services router or Ethernet switch to the MSAN), the outer tag determines which port to forward traffic. The forwarding device then uses the VLAN pop function on this tag before forwarding the traffic with a single tag. The reverse process occurs for upstream traffic.

VLAN stacking is not necessary for S-VLANs or M-VLANs. However, for the hybrid (C-VLAN and M-VLAN) model, the Ethernet switch or services router must be able to pop or push tags onto C-VLAN traffic while not modifying M-VLAN packets.

Related Documentation

- Static Subscriber Interfaces and VLAN Overview on page 6

Dynamic 802.1Q VLAN Overview

You can identify VLANs statically or dynamically. You can also configure a mix of static and dynamic VLANs on the same underlying interface.

For Ethernet, Fast Ethernet, Tri-Rate Ethernet copper, Gigabit Ethernet, 10-Gigabit Ethernet, and aggregated Ethernet interfaces supporting VPLS, the Junos OS supports a subset of the IEEE 802.1Q standard for channelizing an Ethernet interface into multiple logical interfaces. Many hosts can be connected to the same Gigabit Ethernet switch, but they cannot be in the same routing or bridging domain.

To identify VLANs statically, you can reference a static VLAN interface in a dynamic profile. To identify subscribers dynamically, you use a variable to specify an 802.1Q VLAN that is dynamically created when a subscriber accesses the network.

Dynamic VLAN Configuration

You can configure the router to dynamically create VLANs when a client accesses an interface and requests a VLAN ID that does not yet exist. When a client accesses a particular interface, the router instantiates a VLAN dynamic profile that you have associated with the interface. Using the settings in the dynamic profile, the router extracts information about the client from the incoming packet (for example, the interface and unit values), saves this information in the routing table, and creates a VLAN or stacked VLAN ID for the client from a range of VLAN IDs that you configure for the interface.

NOTE: **Dynamic VLAN configuration supports the creation of IPv4 (inet), DHCPv4, IPv6 (inet6), and DHCPv6 VLANs.**

Dynamically configuring VLANs or stacked VLANs requires the following general steps:

1. Configure a dynamic profile for dynamic VLAN or dynamic stacked VLAN creation.

   See *Configuring VLAN Dynamic Profiles*.
2. Associate the VLAN or stacked VLAN dynamic profile with the interface.

   See *Configuring VLAN Interfaces to Use Dynamic Profiles*.
3. Specify the Ethernet packet type that the VLAN dynamic profile accepts.

   See *Configuring Which VLAN Ethernet Packet Types Dynamic Profiles Can Accept*.
4. Define VLAN ranges for use by the dynamic profile when creating VLAN IDs.

   See *Configuring VLAN Ranges for Use with Dynamic Profiles*.

Static Subscriber Interfaces and VLAN Overview

This topic describes the topology for configuring subscriber interfaces over static VLAN interfaces.

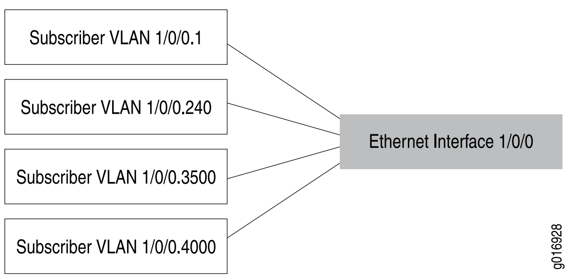
In a dynamic profile, you can configure VLAN subscriber interfaces over the following statically created logical interface types:

- GE—Gigabit Ethernet
- XE—10-Gigabit Ethernet
- AE—Aggregated Ethernet

We recommend that you configure each subscriber on a statically created VLAN.

Figure 1 on page 6 shows an example of subscriber interfaces on an individual VLAN.

Figure 1: VLAN Subscriber Interfaces



You can further separate VLANs on subscriber interfaces by configuring a VLAN interface as the underlying interface for a set of IP demux interfaces.

Related Documentation

- *Configuring a Subscriber Interface with a Static VLAN Interface*
- Subscriber Interfaces and Demultiplexing Overview on page 70

CHAPTER 2

Configuring Dynamic Profiles and Interfaces Used to Create Dynamic VLANs

Configuring a Dynamic Profile Used to Create Single-Tag VLANs

You can configure a dynamic profile for creating stacked 802.1Q VLANs.

Before you begin:

- Configure the dynamic profile.

  See *Configuring a Basic Dynamic Profile*.

To configure a dynamic VLAN profile:

1. Ensure that the VLAN dynamic profile uses the **$junos-interface-ifd-name** variable for the dynamic interface and the **$junos-interface-unit** variable for the interface unit.
2. (Optional) To support dynamic demux interfaces, enable them for IPv4 or IPv6.
   - For IPv4 demux interfaces:

     [edit dynamic-profiles STACKED-VLAN-PROF1 interfaces "$junos-interface-ifd-name" unit "$junos-interface-unit"]
     user@host# **set demux-source inet**

- For IPv6 demux interfaces:

  [edit dynamic-profiles STACKED-VLAN-PROF1 interfaces "$junos-interface-ifd-name" unit "$junos-interface-unit"]
  user@host# **set demux-source inet6**

3. (Optional) To configure the router to respond to any ARP request, specify the proxy-arp statement.

   [edit dynamic-profiles STACKED-VLAN-PROF1 interfaces "$junos-interface-ifd-name" unit "$junos-interface-unit"]
   user@host# **set proxy-arp**

4. Specify the outer VLAN ID variable.

   [edit dynamic-profiles STACKED-VLAN-PROF1 interfaces "$junos-interface-ifd-name" unit "$junos-interface-unit"]
   user@host# **set vlan-tags outer $junos-stacked-vlan-id**

   The variable is dynamically replaced with an outer VLAN ID within the VLAN range specified at the **[interfaces]** hierarchy level.

5. Specify the inner VLAN ID variable.

   [edit dynamic-profiles STACKED-VLAN-PROF1 interfaces "$junos-interface-ifd-name" unit "$junos-interface-unit"]
   user@host# **set vlan-tags inner $junos-vlan-id**

   The variable is dynamically replaced with an inner VLAN ID within the VLAN range specified at the **[interfaces]** hierarchy level.

6. Define the unit family type.

   a. For IPv4 interfaces:

      [edit dynamic-profiles STACKED-VLAN-PROF1 interfaces "$junos-interface-ifd-name" unit "$junos-interface-unit"]
      user@host# **set family inet**

   b. For IPv6 interfaces:

      [edit dynamic-profiles STACKED-VLAN-PROF1 interfaces "$junos-interface-ifd-name" unit "$junos-interface-unit"]
      user@host# **set family inet6**

7. (Optional) Enable IP and MAC address validation for dynamic demux interfaces in a dynamic profile.

   [edit dynamic-profiles STACKED-VLAN-PROF1 interfaces "$junos-interface-ifd-name" unit "$junos-interface-unit" family inet]
   user@host# **set mac-validate loose**

8. Specify the unnumbered address and preferred source address.

   [edit dynamic-profiles STACKED-VLAN-PROF1 interfaces "$junos-interface-ifd-name" unit "$junos-interface-unit" family inet]
   user@host# **set unnumbered-address lo.0 preferred-source-address 192.0.16.1**

Related Documentation

- Configuring an Interface to Use the Dynamic Profile Configured to Create Single-Tag VLANs on page 9

- *Configuring a Basic Dynamic Profile*
- Dynamic 802.1Q VLAN Overview on page 5
- *Dynamic Variables Overview*
- *Junos OS Predefined Variables*

Configuring an Interface to Use the Dynamic Profile Configured to Create Single-Tag VLANs

You configure an interface to use a dynamic profile when the dynamic VLANs are created. The dynamic profile uses the VLAN ranges configured for the interface.

To configure the interface:

1. Access the interface over which you want to create dynamic VLANs.

   ```
   user@host# edit interfaces ge-0/0/0
   ```

2. Access the VLAN range configuration

   ```
   [edit interfaces ge-0/0/0]
   user@host# edit auto-configure vlan-ranges
   ```

3. Specify the dynamic profile used to create VLANs.

   ```
   [edit interfaces ge-0/0/0 auto-configure vlan-ranges]
   user@host# edit dynamic-profile VLAN-PROF-1
   ```

4. Specify the VLAN Ethernet packet type the VLAN dynamic profile accepts.

   inet and **dhcp-v4** for IPv4 packets, **inet6** and **dhcp-v6** for IPv6 packets, and **pppoe** for PPP packets are supported.

   ```
   [edit interfaces ge-0/0/0 auto-configure vlan-ranges dynamic-profile VLAN-PROF-1]
   user@host# set accept inet
   ```

5. Specify the VLAN ranges that you want the dynamic profile to use. The following example specifies a lower VLAN ID limit of 3000 and any upper VLAN ID limit (a range from 1 through 4094).

   ```
   [edit interfaces ge-0/0/0 auto-configure vlan-ranges dynamic-profile VLAN-PROF1]
   user@host# set ranges 3000-any
   ```

   NOTE: You can configure multiple VLAN range groups (up to 32 total) on the same physical interface that use different VLAN dynamic profiles.

6. (Optional) Access another VLAN dynamic profile for which you want to configure VLAN ranges. Specify the VLAN ranges that you want the dynamic profile to use. The following example specifies a lower VLAN ID limit of 2000 and any upper VLAN ID limit (a range from 1 through 4094).

   ```
   [edit interfaces ge-0/0/0 auto-configure vlan-ranges]
   user@host# edit dynamic-profile VLAN-PROF2
   user@host# set ranges 2000-any
   ```

Related Documentation

- Configuring a Dynamic Profile Used to Create Single-Tag VLANs on page 7
- Dynamic 802.1Q VLAN Overview on page 5
- *Configuring VLAN Dynamic Profiles*
- *Configuring VLAN Interfaces to Use Dynamic Profiles*

Configuring a Dynamic Profile Used to Create Stacked VLANs

You can configure a dynamic profile for creating stacked 802.1Q VLANs.

Before you begin:

- Configure the dynamic profile.

  See *Configuring a Basic Dynamic Profile*.

To configure a dynamic VLAN profile:

1. Ensure that the VLAN dynamic profile uses the **$junos-interface-ifd-name** variable for the dynamic interface and the **$junos-interface-unit** variable for the interface unit.
2. (Optional) To support dynamic demux interfaces, enable them for IPv4 or IPv6.
   - For IPv4 demux interfaces:

     ```
     [edit dynamic-profiles VLAN-PROF1 interfaces "$junos-interface-ifd-name" unit
       "$junos-interface-unit"]
     user@host# set demux-source inet
     ```
   - For IPv6 demux interfaces:

     ```
     [edit dynamic-profiles VLAN-PROF1 interfaces "$junos-interface-ifd-name" unit
       "$junos-interface-unit"]
     user@host# set demux-source inet6
     ```
3. (Optional) To configure the router to respond to any ARP request, specify the proxy-arp statement.

   ```
   [edit dynamic-profiles VLAN-PROF1 interfaces "$junos-interface-ifd-name" unit
     "$junos-interface-unit"]
   user@host# set proxy-arp
   ```
4. Specify that you want to use dynamic VLAN IDs in the dynamic profile. You can configure the dynamic profile to create a single-tag VLAN using only standard tag protocol identifier (TPID) values (0x8100) or to create a VLAN using any TPID value.
   - To configure the dynamic profile to create single-tag VLANs that accept only standard TPID values (a TPID value of 0x8100):

     ```
     [edit dynamic-profiles VLAN-PROF1 interfaces "$junos-interface-ifd-name" unit
       "$junos-interface-unit"]
     user@host# set vlan-id $junos-vlan-id
     ```

     When the dynamic profile is instantiated, the variable is dynamically replaced with a VLAN ID within the VLAN range specified at the **[interfaces]** hierarchy level.

- To configure the dynamic profile to create single-tag VLANs that accept any TPID value:

  [edit dynamic-profiles VLAN-PROF1 interfaces "$junos-interface-ifd-name" unit "$junos-interface-unit"]
  user@host# set vlan-tags outer $junos-vlan-id

  The variable is dynamically replaced with both the TPID value and a VLAN ID within the VLAN range specified at the **[interfaces]** hierarchy level.

5. Define the unit family type.

   a. For IPv4 interfaces:

      [edit dynamic-profiles VLAN-PROF1 interfaces "$junos-interface-ifd-name" unit "$junos-interface-unit"]
      user@host# set family inet

   b. For IPv6 interfaces:

      [edit dynamic-profiles VLAN-PROF1 interfaces "$junos-interface-ifd-name" unit "$junos-interface-unit"]
      user@host# set family inet6

6. (Optional) Enable IP and MAC address validation for dynamic demux interfaces in a dynamic profile.

   [edit dynamic-profiles VLAN-PROF1 interfaces "$junos-interface-ifd-name" unit "$junos-interface-unit" family inet]
   user@host# set mac-validate loose

7. Specify the unnumbered address and preferred source address.

   [edit dynamic-profiles VLAN-PROF1 interfaces "$junos-interface-ifd-name" unit "$junos-interface-unit" family inet]
   user@host# set unnumbered-address lo.0 preferred-source-address 192.0.16.1

Related Documentation

- Configuring an Interface to Use the Dynamic Profile Configured to Create Stacked VLANs on page 11
- *Configuring a Basic Dynamic Profile*
- Dynamic 802.1Q VLAN Overview on page 5
- *Dynamic Variables Overview*
- *Junos OS Predefined Variables*

Configuring an Interface to Use the Dynamic Profile Configured to Create Stacked VLANs

You configure an interface to use a dynamic profile when the dynamic VLANs are created. The dynamic profile uses the VLAN ranges configured for the interface.

To configure the interface:

1. Access the interface over which you want to create dynamic VLANs.

user@host# **edit** interfaces **ge-0/0/0**

2. Specify that this interface is for use with stacked VLAN ranges.

   [edit interfaces ge-0/0/0]
   user@host# **set** stacked-vlan-tagging

3. Access the VLAN range configuration

   [edit interfaces ge-0/0/0]
   user@host# **edit** auto-configure stacked-vlan-ranges

4. Specify the dynamic profile used to create VLANs.

   [edit interfaces ge-0/0/0 auto-configure vlan-ranges]
   user@host# **edit** dynamic-profile **STACKED-VLAN-PROF1**

5. Specify the VLAN Ethernet packet type the VLAN dynamic profile accepts.

   inet and **dhcp-v4** for IPv4 packets, **inet6** and **dhcp-v6** for IPv6 packets, and **pppoe** for PPP packets are supported.

   [edit interfaces ge-0/0/0 auto-configure vlan-ranges dynamic-profile
   STACKED-VLAN-PROF1]
   user@host# **set** accept **inet**

6. Specify the outer and inner stacked VLAN ranges that you want the dynamic profile to use. The following example specifies an outer stacked VLAN ID range from 2000 through 4000 and an inner stacked VLAN ID range of **any** (enabling a range from 1 through 4094 for the inner stacked VLAN ID).

   [edit interfaces ge-0/0/0 auto-configure vlan-ranges dynamic-profile VLAN-PROF1]
   user@host# **set** ranges **2000-4000,any**

   NOTE: You can configure multiple dynamic profile associations (up to 32) with different VLAN range groups on each physical interface.

7. (Optional) Access another VLAN dynamic profile for which you want to configure VLAN ranges.

   [edit interfaces ge-0/0/0 auto-configure vlan-ranges]
   user@host# **edit** dynamic-profile **VLAN-PROF2**

8. (Optional) Specify the outer and inner stacked VLAN ranges that you want the dynamic profile to use. The following example specifies an outer stacked VLAN ID range from 3001 through 4000 and an inner stacked VLAN ID range of **any** (enabling a range from 1 through 4094 for the inner stacked VLAN ID).

   [edit interfaces ge-0/0/0 auto-configure vlan-ranges dynamic-profile VLAN-PROF2]
   user@host# **set** ranges **3001-4000,any**

Related Documentation

- Configuring a Dynamic Profile Used to Create Stacked VLANs on page 10
- Dynamic 802.1Q VLAN Overview on page 5

- *Configuring VLAN Dynamic Profiles*

Configuring Interfaces to Support Both Single and Stacked VLANs

You can configure VLANs to support simultaneous transmission of 802.1Q VLAN single-tag and stacked frames on logical interfaces on the same Ethernet port, and on pseudowire logical interfaces.

Junos VLAN IDs for single-tag VLANs are equivalent to the outer tags used for stacked (dual-tag) VLANs. When configuring mixed (flexible) VLANs, any overlap on single-tag VLAN IDs and stacked VLAN outer tag values is supported only for dynamic VLANs on MPC line cards. When configuring mixed (flexible) VLANS on DPCE line cards, overlapping single-tag VLAN IDs and stacked VLAN outer tag values is not supported. This means that a dynamically created single-tagged VLAN interface prevents any overlapping stacked VLAN interfaces from being created or a dynamically created stacked VLAN interface prevents any overlapping single-tagged VLAN interfaces from being created.

To configure both VLAN and stacked VLAN ranges:

1. Access the interface over which you want to create dynamic VLANs.

   ```
   user@host# edit interfaces ge-0/0/0
   ```

2. Indicate that this interface is for use with both VLAN and stacked VLAN ranges.

   ```
   [edit interfaces ge-0/0/0]
   user@host# set flexible-vlan-tagging
   ```

3. Define interface automatic configuration values.

   ```
   [edit interfaces ge-0/0/0]
   user@host# edit auto-configure
   ```

4. Specify that you want to modify VLAN ranges.

   ```
   [edit interfaces ge-0/0/0 auto-configure]
   user@host# edit vlan-ranges
   ```

5. Access the VLAN dynamic profile for which you want to configure VLAN ranges.

   ```
   [edit interfaces ge-0/0/0 auto-configure vlan-ranges]
   user@host# edit dynamic-profile VLAN-PROF1
   ```

6. Specify the VLAN ranges that you want the dynamic profile to use. The following example specifies a lower VLAN ID limit of 2000 and an upper VLAN ID limit of 3000.

   ```
   [edit interfaces ge-0/0/0 auto-configure vlan-ranges dynamic-profile VLAN-PROF1]
   user@host# set ranges 2000-3000
   ```

   NOTE: You can configure multiple dynamic profile associations (up to 32) with different VLAN range groups on each physical interface.

7. Specify that you want to modify stacked VLAN ranges.

   ```
   [edit interfaces ge-0/0/0 auto-configure]
   user@host# edit stacked-vlan-ranges
   ```

8. Access the VLAN dynamic profile for which you want to configure VLAN ranges.

```
[edit interfaces ge-0/0/0 auto-configure stacked-vlan-ranges]
user@host# edit dynamic-profile VLAN-PROF2
```

9. Specify the outer and inner stacked VLAN ranges that you want the dynamic profile to use. The following example specifies an outer stacked VLAN ID range from 3001 through 4000 (to avoid overlapping VLAN IDs with single-tag VLANs) and an inner stacked VLAN ID range of **any** (enabling a range from 1 through 4094 for the inner stacked VLAN ID).

```
[edit interfaces ge-0/0/0 auto-configure stacked-vlan-ranges dynamic-profile
  VLAN-PROF2]
user@host# set ranges 3001-4000,any
```

NOTE: You can configure multiple dynamic profile associations (up to 32) with different VLAN range groups on each physical interface.

Related Documentation

- Configuring an Interface to Use the Dynamic Profile Configured to Create Single-Tag VLANs on page 9
- Configuring an Interface to Use the Dynamic Profile Configured to Create Stacked VLANs on page 11
- Dynamic 802.1Q VLAN Overview on page 5
- *Configuring VLAN Dynamic Profiles*

Overriding the Dynamic Profile Used for an Individual VLAN

You can override dynamic profile assignment to individual VLANs that are already part of a previously defined VLAN range. This functionality provides a type of exception to an assigned VLAN range. It enables you to configure individual VLAN IDs to use a different dynamic profile from the one assigned to the VLAN range that includes the individual VLAN ID.

To configure dynamic profile override for a specific VLAN:

1. Access the interface on which you want to create a dynamic profile override.

```
user@host# edit interfaces ge-0/0/0
```

2. Access the interface automatic configuration hierarchy.

```
[edit interfaces ge-0/0/0]
user@host# edit auto-configure
```

3. Access either the single-tagged or dual-tagged (stacked) VLAN ranges that you want to modify.

```
[edit interfaces ge-0/0/0 auto-configure]
user@host# edit vlan-ranges
```

or

```
[edit interfaces ge-0/0/0 auto-configure]
```

user@host# **edit** stacked-vlan-ranges

4. Define the **override** statement along with the VLAN tag that you want to override and the dynamic profile that you want to use when overriding the specified VLAN tag.

   [edit interfaces ge-0/0/0 auto-configure vlan-ranges]
   user@host# **set** override **tag 20 dynamic-profile NewProfile**

   or

   [edit interfaces ge-0/0/0 auto-configure stacked-vlan-ranges]
   user@host# **set** override **tag 20 dynamic-profile NewProfile**

Configuring a VLAN Dynamic Profile That Associates VLANs with Separate Routing Instances

You can configure a VLAN dynamic profile that dynamically creates underlying VLAN interfaces and associates these interfaces with statically created routing instances. The VLAN interface is created for a specific routing instance as defined by VSA 26–1 (Virtual-Router) on the AAA server (for example, RADIUS server).

To configure a dynamic VLAN profile to use routing instances when creating VLANs, add the routing instance configuration to your dynamic profile:

1. Access the dynamic profile.

   [edit]
   user@host# **edit dynamic-profiles VLAN_PROFILE_RI**

2. Specify that you want to dynamically associate the profile with routing instances.

   [edit dynamic-profiles VLAN_PROFILE_RI]
   user@host# **edit routing-instances $junos-routing-instance**

3. Define the routing instance **interface** statement with the internal **$junos-interface-name** variable used by the router to match the interface name of the receiving interface.

   [edit dynamic-profiles VLAN_PROFILE_RI routing-instances “$junos-routing-instance”]
   user@host# **set interface $junos-interface-name**

4. Define the dynamic profile **interfaces** statement with the internal **$junos-interface-ifd-name** variable.

   [edit dynamic-profiles VLAN_PROFILE_RI]
   user@host# **edit interfaces $junos-interface-ifd-name**

5. Define the **unit** statement with the internal **$junos-interface-unit** variable used by the router to generate a unit value for the interface.

   [edit dynamic-profiles VLAN_PROFILE_RI interfaces "$junos-interface-ifd-name"]
   user@host# **edit unit $junos-interface-unit**

Related Documentation

- *Configuring a Basic Dynamic Profile*
- Dynamic 802.1Q VLAN Overview on page 5
- *Dynamic Variables Overview*
- *Junos OS Predefined Variables*

- *Configuring Frames with Particular TPIDs to Be Processed as Tagged Frames*
- Configuring Dynamic Authentication for VLAN Interfaces on page 20

Automatically Removing VLANs with No Subscribers

You can always clear or delete subscriber VLANs manually. However, you can also configure the interface to automatically remove dynamic subscriber VLANs when no client sessions (for example, DHCP or PPPoE) exist on the VLAN.

When configuring automatic removal of dynamic subscriber VLANs, keep the following in mind:

- You can configure automatic VLAN removal only on individual physical interfaces. You cannot configure the feature globally.
- Automatic VLAN removal is not supported for use on Layer 2 Wholesale interfaces. See *Layer 2 and Layer 3 Wholesale Overview*.
- PPPoE subscriber interfaces require the use of a dynamic profiles when configured over dynamic VLANS. However, dynamic profiles are not required for use with DHCP subscriber interfaces that use underlying dynamic VLANs. Because the remove-when-no-subscribers functionality triggers when no dynamic client sessions exist on a dynamic VLAN, automatic removal of underlying dynamic VLANs is not supported when DHCP subscriber interfaces are not created using dynamic profiles.
- The **maintain-subscriber** statement and **remove-when-no-subscribers** statement are mutually exclusive. When the router is configured to maintain subscribers, you cannot also specify that dynamically configured VLAN interfaces are removed when no subscribers exist.
- If PPPoE subscriber session lockout is also configured, the router does not remove the unused subscriber VLAN until the lockout time has expired for each client undergoing lockout on the underlying interface.

To configure automatic removal of subscriber VLANs when no client sessions exist on the VLAN:

1. Access the interface for which you want to enable automatic removal of subscriber VLANs.

   user@host# **edit interfaces ge-1/1/1**

2. Access the interface automatic configuration hierarchy.

   [edit interfaces ge-1/1/1]
   user@host# **edit auto-configure**

3. Enable subscriber VLAN removal with the **remove-when-no-subscribers** statement.

   [edit interfaces ge1/1/1 auto-configure]
   user@host# **set remove-when-no-subscribers**

Related Documentation

- Dynamic 802.1Q VLAN Overview on page 5
- *Configuring VLAN Interfaces to Use Dynamic Profiles*
- *Layer 2 and Layer 3 Wholesale Overview*
- *Layer 2 Wholesale Network Topology Overview*
- PPPoE Subscriber Session Lockout Overview on page 191

Verifying and Managing Dynamic VLAN Configuration

Purpose View or clear information about dynamic VLANs and stacked VLANs.

Action

- To display subscriber dynamic VLAN information:

  user@host>**show subscribers detail**

- To display interface-specific output for dynamic VLANs:

  user@host>**show interfaces *interface-name***

- To clear the binding state of dynamic VLAN interfaces:

  user@host> **clear auto-configuration interfaces**

Related Documentation

- CLI Explorer

CHAPTER 3

Configuring Subscriber Authentication for Dynamic VLANs

Configuring an Authentication Password for VLAN or Stacked VLAN Ranges

You can specify an authentication password for dynamically created VLAN or stacked VLAN interfaces at the **[edit interfaces *interface-name* auto-configure vlan-ranges authentication]** or **[edit interfaces *interface-name* auto-configure stacked-vlan-ranges authentication]** hierarchy level. This password is sent to the external AAA authentication server for subscriber authentication.

NOTE: You must configure the username-include statement to enable the use of authentication. The password (Interfaces) statement is not required and does not cause the interface to use authentication if the username-include statement is not included.

To configure an authentication password:

1. Access the interface over which you want to create dynamic VLANs.

   user@host# **edit interfaces ge-0/0/0**

2. Edit the VLAN **auto-configure** stanza.

   [edit interfaces ge-0/0/0]
   user@host# **edit auto-configure**

3. Edit the **vlan-ranges** or **stacked-vlan-ranges** stanza.

```
[edit interfaces ge-0/0/0 auto-configure]
user@host# edit vlan-ranges
```

or

```
[edit interfaces ge-0/0/0 auto-configure]
user@host# edit stacked-vlan-ranges
```

4. Edit the VLAN **authentication** stanza.

```
[edit interfaces ge-0/0/0 auto-configure vlan-ranges]
user@host# edit authentication
```

5. Specify a password that is sent to the external AAA authentication server for subscriber authentication.

```
[edit interfaces ge-0/0/0 auto-configure vlan-ranges]
user@host# set password (Interfaces) PSSWD1
```

Related Documentation

- Configuring Dynamic Authentication for VLAN Interfaces on page 20

Configuring Dynamic Authentication for VLAN Interfaces

You can use dynamic profiles, in conjunction with RADIUS, to dynamically create logical VLAN interfaces in the default logical system and in a specified routing instance. As DHCP clients in the same VLAN become active, corresponding interfaces are assigned to any specified routing instances. You can also dynamically create an underlying VLAN interface for incoming subscribers, associate interfaces created on this VLAN with the default logical system and a specified routing instance, and define RADIUS authentication values for the dynamically created interfaces.

Before you configure dynamic VLAN authentication, configure DHCP Local Server or DHCP Relay over which you want the dynamic VLAN interfaces to function.

For information about DHCP Local Server or DHCP Relay, see:

- *Extended DHCP Local Server Overview*
- *Extended DHCP Relay Agent Overview*

NOTE: You can also configure dynamically created VLAN interfaces over PPP or PPPoE interfaces. For information about how to configure PPP or PPPoE, see *Dynamic Profiles for PPP Subscriber Interfaces Overview* or "Subscriber Interfaces and PPPoE Overview" on page 141.

To configure dynamic authentication for dynamically created VLAN interfaces:

1. Configure an access profile that contains the appropriate accounting order, authentication order, and server access values.

   For information about how to configure an access profile, RADIUS accounting, RADIUS statistics, and how to define RADIUS server access, see:

   - *Configuring an Access Profile for Subscriber Management*
   - *Specifying the Authentication and Accounting Methods for Subscriber Access*
   - *Configuring Per-Subscriber Session Accounting*
   - *Configuring Router or Switch Interaction with RADIUS Servers*

2. Configure a dynamic profile that uses the default logical system and creates specific routing instances to contain dynamically created VLAN interfaces.

   See "Configuring a VLAN Dynamic Profile That Associates VLANs with Separate Routing Instances" on page 15.

3. Define the VLAN physical interface for automatic configuration.

   See the following topics:

   - *Enabling VLAN Tagging*
   - *Configuring Which VLAN Ethernet Packet Types Dynamic Profiles Can Accept*
   - *Configuring VLAN Ranges for Use with Dynamic Profiles*
   - Configuring an Authentication Password for VLAN or Stacked VLAN Ranges on page 19
   - Configuring VLAN Interface Username Information for AAA Authentication on page 22

4. Associate an access profile to the VLAN interface.

   See *Attaching Access Profiles*.

5. Associate a dynamic profile to the VLAN interface.

   See *Configuring VLAN Interfaces to Use Dynamic Profiles*.

Related Documentation

- Dynamic 802.1Q VLAN Overview on page 5

Configuring Subscriber Packet Types to Trigger VLAN Authentication

By default, VLAN authentication is triggered by any of the packet types specified with the **accept** statement in the dynamic profile that instantiates the VLAN and subscriber interfaces. For certain business cases, you may want a more generic dynamic profile that includes several packet types, but in some situations want the VLAN to be authenticated for only a subset of your customers. You can use the **packet-types** statement to specify the desired subset.

To limit triggering of VLAN authentication to a subset of accepted packet types:

- Specify one or more packet types that you want to trigger VLAN authentication.

  [edit interfaces *interface-name* auto-configure vlan-ranges authentication]
  user@host# **set** packet-types **[*packet-type*]**

For example, the following partial configuration shows how to specify that IP, IPv6, and PPPoE packet types trigger the creation of autoconfigured, single-tagged VLANs, but only IP and IPv6 packets trigger authentication:

1. Access the VLAN dynamic profile for which you want to configure VLAN ranges.

   [edit interfaces ge-0/0/0 auto-configure vlan-ranges]
   user@host# **edit** dynamic-profile **VLAN-PROF-1**

2. Specify the VLAN ranges for the VLAN dynamic profile.

   [edit interfaces ge-0/0/0 auto-configure vlan-ranges dynamic-profile VLAN-PROF-1]
   user@host# **set** ranges **any**

3. Specify the VLAN packet types accepted by the VLAN dynamic profile.

   [edit interfaces ge-0/0/0 auto-configure vlan-ranges dynamic-profile VLAN-PROF-1]
   user@host# **set** accept **[inet inet6 pppoe]**

4. Specify the subset of those packet types that you want to trigger VLAN authentication.

   [edit interfaces ge-0/0/0 auto-configure vlan-ranges authentication]
   user@host# **set** packet-types **[inet inet6]**

Related Documentation

- *Configuring VLAN Dynamic Profiles*
- *Configuring VLAN Interfaces to Use Dynamic Profiles*
- *Configuring the VLAN Ethernet Packet Type for Single-Tag VLAN Dynamic Profiles*
- *Configuring the VLAN Ethernet Packet Type for Stacked VLAN Dynamic Profiles*

Configuring VLAN Interface Username Information for AAA Authentication

You can define interface information that is included in the username that is subsequently passed to the external AAA authentication service (for example, RADIUS) when creating dynamic VLANs or stacked VLANs. The AAA authentication service uses this information to authenticate the VLAN or stacked VLAN physical interface. After the interface is authenticated, the AAA service can send the required routing instance values to the system for use in dynamically creating VLAN or stacked VLAN interfaces.

NOTE: The following example configures username information on VLANs. However, you can also configure dynamic authentication on stacked VLANs by configuring the same statements at the **[edit interfaces *interface-name* auto-configure stacked-vlan-ranges authentication]** hierarchy level.

To configure VLAN interface username information:

1. Access the interface over which you want to configure username information.

   user@host# **edit interfaces ge-0/0/0**

2. Edit the **auto-configure** stanza.

   [edit interfaces ge-0/0/0]
   user@host# **edit auto-configure**

3. Edit the **vlan-ranges** stanza.

   [edit interfaces ge-0/0/0 auto-configure]
   user@host# **edit vlan-ranges**

4. Edit the **authentication** stanza.

   [edit interfaces ge-0/0/0 auto-configure vlan-ranges]
   user@host# **edit authentication**

5. Edit the **username-include** stanza.

   [edit interfaces ge-0/0/0 auto-configure vlan-ranges]
   user@host# **edit username-include**

6. Specify the username statements that you want the AAA authentication service to use to authenticate the username.

   [edit interfaces ge-0/0/0 auto-configure vlan-ranges authentication username-include]
   user@host# **set delimiter**

Related Documentation

- Configuring Dynamic Authentication for VLAN Interfaces on page 20
- Using DHCP Option 82 Suboptions in Authentication Usernames for Autosense VLANs on page 23

Using DHCP Option 82 Suboptions in Authentication Usernames for Autosense VLANs

You can specify the Option 82 suboptions that are concatenated with the username during the authentication process for autosense VLANs. The option 82 value used in creating the username is based on the option 82 value that is encoded in the incoming DHCP discover packet.

You can specify either, both, or neither of the Agent Circuit ID (suboption 1) and the Agent Remote ID (suboption 2). If you specify both, the Agent Circuit ID is supplied first, followed by a delimiter, and then the Agent Remote ID. If you specify that neither suboption is supplied, the raw payload of Option 82 from the PDU is concatenated to the username. The use of Option 82 suboptions is supported for DHCPv4 discover packets only.

Related Documentation

- Configuring VLAN Interface Username Information for AAA Authentication on page 22

Using DHCP Option 18 and Option 37 in Authentication Usernames for DHCPv6 Autosense VLANs

For DHCPv4, Option 82 has suboptions containing the ACI and ARI that are concatenated with the username during the authentication process for autosense (dynamic) VLANs. For DHCPv6, the relay agent uses Options 18 and Option 37 to convey the ACI and ARI, respectively. You can include these options in the username to generate unique usernames that identify subscribers for authentication in DHCPv6 dynamic VLANs.

A DHCPv6 Solicit message encapsulated with a Relay-Forward message header and one without the Relay-Forward message header are eligible for dynamic VLAN creation when you configure the DHCPv6 packet type for autosensing. Options 18 and Option 37 are provided in the Relay-Forward message header and are extracted only from this header and not from the options within the DHCPv6 Solicit message. In addition, if the DHCPv6 Solicit message is encapsulated in multiple Relay-Forward message headers, only the option values from the innermost Relay-Forward message header are used for username authentication. If these options are sent by the client or DHCPv6 relay agent, and if dynamic VLAN authentication is configured to use these options in the username, then the options are included in the username for authentication. If either of these options is not sent by the client or DHCPv6 relay agent, or if the dynamic VLAN authentication is not configured to use the option in the username, the username is constructed without the option.

To include Option 18 or Option 37 in the username for DHCPv6 dynamic VLANs, include the **option-18** and **option-37** statements at the **[edit interfaces *interface-name* auto-configure vlan-ranges authentication username-include]** hierarchy level. To include Options 18 or Option 37 in the username for stacked VLANs, include **option-18** and **option-37** statements at the **[edit interfaces *interface-name* auto-configure stacked-vlan-ranges authentication username-include]** hierarchy level.

Related Documentation

- Configuring VLAN Interface Username Information for AAA Authentication on page 22
- **username-include on page 590**
- **option-18 on page 524**
- **option-37 on page 525**
- *Configuring Which VLAN Ethernet Packet Types Dynamic Profiles Can Accept*

CHAPTER 4

Configuring VLANs for Households or Individual Subscribers Using ACI-Based Dynamic VLANs

Agent Circuit Identifier-Based Dynamic VLANs Overview

You can configure the router to create dynamic virtual LAN (VLAN) subscriber interfaces for Dynamic Host Configuration Protocol (DHCP) and Point-to-Point Protocol over Ethernet (PPPoE) subscribers based on agent circuit identifier (ACI) information. To use ACI-based dynamic VLAN subscriber interfaces, you must configure them on Modular Port Concentrators/Modular Interface Cards (MPCs/MICs) that face the access side of the network in an MX Series router.

This overview covers the following topics:

VLAN Architectures and Subscriber Identification

The following VLAN architectures defined in the DSL Forum Technical Report (TR)-101, Migration to Ethernet-Based DSL Aggregation (April 2006), use different methods to uniquely identify subscribers in Ethernet-based subscriber access networks:

- 1:1 access model using customer VLANs

  Configurations that use the 1:1 access model uniquely identify subscribers by means of VLAN encapsulation; that is, by using the VLAN ID and stacked VLAN (S-VLAN) ID. Subscriber packets received from the access node (such as a digital subscriber line access multiplexer, or DSLAM) that are either single-tagged with a VLAN ID or double-tagged with both an S-VLAN ID and a VLAN ID are examples of 1:1 VLAN configurations because they provide a one-to-one correspondence between an individual subscriber and the VLAN encapsulation.

  In the 1:1 VLAN architecture, each customer premises equipment (CPE) or subscriber network has its own dedicated Layer 2 path to the router. Each subscriber network is separated by a customer VLAN (C-VLAN) that is dedicated to a particular customer. The services for each customer are transmitted from the router to the access node by means of that customer's C-VLAN.

  The ability to uniquely identify subscribers by means of VLAN encapsulation facilitates delivery of services such as authentication, authorization, and accounting (AAA); class of service (CoS); and filters (policers) to subscribers in a 1:1 VLAN configuration.

- N:1 access model using service VLANs

  Configurations that use the N:1 access model do not uniquely identify subscribers by means of VLAN encapsulation. Instead, these configurations identify subscribers by means of the agent circuit identifier (ACI) information present in DHCP and PPPoE control packets. Subscriber packets received from the access node that are either single-tagged with the same VLAN ID for a group of subscribers or untagged are examples of N:1 VLAN configurations because they provide a many-to-one correspondence between individual subscribers and the VLAN encapsulation.

  In the N:1 VLAN architecture, a service such as video, voice, or data is typically routed to a particular VLAN instead of having multiple services share a single VLAN, as is the case with the 1:1 VLAN architecture. Such VLANs, often referred to as service VLANs, enable service providers to route different services to different routers to functionally separate network services and reduce network complexity.

  Because a VLAN in an N:1 configuration corresponds to a service rather than an individual subscriber, the router uses ACI information in DHCP and PPPoE control packets instead of VLAN encapsulation to uniquely identify subscribers and facilitate application of subscriber-based services.

ACI-Based Dynamic VLANs and Agent Circuit Identifier Interface Sets

For single-tagged, double-tagged, or untagged N:1 configurations that do not use VLAN encapsulation to uniquely identify subscribers, you can configure the router to create dynamic VLAN subscriber interfaces for DHCP and PPPoE subscribers based on ACI information. ACI-based dynamic VLANs uniquely identify subscribers on the router and facilitate application of subscriber-based services, such as CoS and interface-shared filters, to all subscribers that originate from a single household and share the same ACI information.

When you configure an ACI-based dynamic VLAN, the router examines the DHCP and PPPoE control packets to extract the ACI information in order to build a unique dynamic VLAN subscriber interface. The agent-circuit-identifier value is a string that uniquely identifies the subscriber's access node and the digital subscriber line (DSL) on the access node. For DHCP traffic, the agent-circuit-identifier string is in the DHCP option 82 field of DHCP messages. For PPPoE traffic, the agent-circuit-identifier string is in the DSL Forum Agent-Circuit-ID VSA [26-1] of PPPoE Active Discovery Initiation (PADI) and PPPoE Active Discovery Request (PADR) control packets.

Configuring ACI-based dynamic VLAN subscriber interfaces is particularly useful in configurations with multiple DHCP and PPPoE subscriber sessions per household. Because DHCP and PPPoE control traffic sent to the router from the same household has the same unique agent-circuit-identifier string, the router groups these DHCP and PPPoE subscriber interfaces in the same ACI interface set. An *ACI interface set* is a logical collection of subscriber interfaces that originate at the same household or on the same access-loop port. Grouping subscriber interfaces into ACI interface sets enables unique subscriber identification and facilitates application of subscriber-based services, such as class of service (CoS) and interface-shared filters, on a per-household basis.

Related Documentation

- Agent Circuit Identifier-Based Dynamic VLANs Components Overview on page 27
- Configuring Dynamic VLANs Based on Agent Circuit Identifier Information on page 29
- Verifying and Managing Agent Circuit Identifier-Based Dynamic VLAN Configuration on page 38
- Clearing Agent Circuit Identifier Interface Sets on page 39

Agent Circuit Identifier-Based Dynamic VLANs Components Overview

You can configure ACI-based dynamic VLAN subscriber interfaces on Modular Port Concentrators/Modular Interface Cards (MPCs/MICs) that face the access side of the network in an MX Series router.

This overview describes the components of an ACI-based dynamic VLAN configuration, from top to bottom of the interface stack:

- ACI-Based Dynamic Subscriber Interface on page 27
- Dynamic ACI Interface Set on page 28
- Dynamic or Static Underlying VLAN Interface on page 28
- Static Physical Interface on page 28

ACI-Based Dynamic Subscriber Interface

You must create a dynamic profile to define either a dynamic PPPoE subscriber interface for PPPoE subscriber sessions, or a dynamic IP demultiplexer (IP demux) subscriber interface for DHCP subscriber sessions. The router automatically creates (instantiates) the subscriber interface when a DHCP or PPPoE subscriber logs in on the associated underlying VLAN interface associated with the dynamic profile that defines the ACI interface set.

Dynamic ACI Interface Set

The dynamic ACI interface set, which is the primary component of an ACI-based dynamic VLAN configuration, groups the DHCP and PPPoE subscriber sessions that belong to a particular household and share a common unique agent-circuit-identifier value. The router creates one ACI interface set per household to facilitate application of subscriber-based services, such as CoS and interface-shared filters, to all subscribers in the household.

You must create a dynamic profile to define the ACI interface set, which is represented in the profile by the Junos OS predefined dynamic variable **$junos-interface-set-name**. When a DHCP or PPPoE subscriber accesses the router on a particular interface, the router obtains the agent-circuit-identifier information from the DHCP or PPPoE control packets transmitted on that interface and dynamically creates the ACI interface set when the first subscriber from that household logs in.

Dynamic or Static Underlying VLAN Interface

After you define the ACI interface set, you must configure the underlying VLAN interface to enable creation of dynamic VLAN subscriber interfaces based on ACI information. You can configure the underlying VLAN interface either dynamically (with a dynamic profile) or statically.

ACI-based dynamic VLAN configurations support the following underlying VLAN interface types:

- Gigabit Ethernet
- 10-Gigabit Ethernet
- VLAN demux (demux0)

NOTE: When you configure an underlying VLAN interface to support creation of ACI-based dynamic VLANs, we recommend that you use this underlying interface only for subscriber interfaces that contain agent-circuit-identifier information in their DHCP or PPPoE control packets. If the router receives DHCP or PPPoE control packets without agent-circuit-identifier information on an underlying VLAN interface configured for ACI-based dynamic VLANs, the associated subscriber interfaces might not instantiate successfully.

Static Physical Interface

ACI-based dynamic VLAN configurations support the following physical interface types:

- Gigabit Ethernet
- 10-Gigabit Ethernet

- Aggregated Ethernet

Related Documentation

- Agent Circuit Identifier-Based Dynamic VLANs Overview on page 25
- Configuring Dynamic VLANs Based on Agent Circuit Identifier Information on page 29
- Verifying and Managing Agent Circuit Identifier-Based Dynamic VLAN Configuration on page 38
- Clearing Agent Circuit Identifier Interface Sets on page 39

Configuring Dynamic VLANs Based on Agent Circuit Identifier Information

On MX Series routers with Modular Port Concentrators/Modular Interface Cards (MPCs/MICs) that face the access side of the network, you can configure dynamic VLAN subscriber interfaces based on agent circuit identifier (ACI) information, also known as *ACI-based dynamic VLANs*, for DHCP and PPPoE subscribers. To do so, you create an *ACI interface set*, which is a logical collection of subscriber interfaces that originate at the same household or on the same access-loop port, and then reference the ACI interface set in the dynamic profile for a PPPoE or IP demultiplexing (IP demux) logical subscriber interface.

Grouping subscriber interfaces into ACI interface sets to create ACI-based dynamic VLANs facilitates application of subscriber-based services, such as class of service (CoS) and interface-shared filters, to all of the subscriber interfaces from the same household.

Before you begin:

1. Configure the underlying physical interface for single-tag VLANs or stacked (dual-tag) VLANs.

   See the following topics:

   - *802.1Q VLANs Overview*
   - *Configuring VLAN Dynamic Profiles*
   - *Configuring VLAN Interfaces to Use Dynamic Profiles*

2. Create a dynamic profile that defines the logical subscriber interface.

   See the following topics:

   - *Configuring a Basic Dynamic Profile*
   - Configuring Dynamic PPPoE Subscriber Interfaces Using Dynamic Profiles on page 145
   - Configuring Dynamic Subscriber Interfaces Using IP Demux Interfaces in Dynamic Profiles on page 77

To configure a dynamic VLAN subscriber interface based on ACI information:

1. Create a dynamic profile that defines the dynamic ACI interface set.

   See “Defining Agent Circuit Identifier Interface Sets” on page 31.

2. (Optional) Include attributes for PPPoE, CoS, and interface-shared filters in the dynamic profile for the ACI interface set.

   See “Defining Agent Circuit Identifier Interface Sets” on page 31.

3. (Optional) In the dynamic profile for the ACI interface set, configure the router to use the Actual-Data-Rate-Downstream VSA [26-130] or Access-Loop-Encapsulation VSA [26-144] value in PPPoE control packets to adjust CoS shaping-rate and overhead-accounting attributes at a per-household level.

   See *Adjusting the CoS Shaping Rate and Overhead Accounting Parameters for Agent Circuit Identifier-Based Dynamic VLANs*.

4. Dynamically or statically configure the underlying VLAN logical interface to enable dynamic subscriber interface creation based on ACI information.

   - For dynamic underlying VLAN interfaces, see “Configuring Dynamic Underlying VLAN Interfaces to Use Agent Circuit Identifier Information” on page 33.
   - For static underlying VLAN interfaces, see “Configuring Static Underlying VLAN Interfaces to Use Agent Circuit Identifier Information” on page 35.

5. Associate the dynamic ACI interface set with the dynamic PPPoE or dynamic IP demux logical subscriber interface.

   See “Configuring Dynamic VLAN Subscriber Interfaces Based on Agent Circuit Identifier Information” on page 36.

6. (Optional) In the dynamic profile for the PPPoE (**pp0**) subscriber interface, configure the router to use the Actual-Data-Rate-Downstream VSA [26-130] or Access-Loop-Encapsulation VSA [26-144] value in PPPoE control packets to adjust CoS shaping-rate and overhead-accounting attributes at a per-subscriber level.

   See *Adjusting the CoS Shaping Rate and Overhead Accounting Parameters for Agent Circuit Identifier-Based Dynamic VLANs*.

7. (Optional) Verify the ACI-based dynamic VLAN subscriber interface configuration.

   See “Verifying and Managing Agent Circuit Identifier-Based Dynamic VLAN Configuration” on page 38.

8. (Optional) Clear the ACI interface set from the router when the interface set no longer has any active subscriber sessions.

   See “Clearing Agent Circuit Identifier Interface Sets” on page 39.

Related Documentation

- Agent Circuit Identifier-Based Dynamic VLANs Overview on page 25
- Agent Circuit Identifier-Based Dynamic VLANs Components Overview on page 27
- *Agent Circuit Identifier-Based Dynamic VLANs Bandwidth Management Overview*

Defining Agent Circuit Identifier Interface Sets

To configure the router to create dynamic VLAN subscriber interfaces for DHCP and PPPoE subscribers based on agent circuit identifier (ACI) information, you must create a dynamic ACI interface set. An *ACI interface set* is a logical collection of subscriber interfaces that originate at the same household or on the same access-loop port.

Because DHCP and PPPoE traffic sent to the router from the same household carries the same ACI value in DHCP and PPPoE control packets, the router groups these subscriber interfaces into a single ACI interface set. Grouping subscriber interfaces into ACI interface sets facilitates application of attributes for PPPoE, class of service (CoS), and interface-shared filters to all of the subscriber interfaces from the same household.

To configure an ACI interface set in a dynamic profile:

1. Name the dynamic profile that defines the ACI interface set.

   [edit]
   user@host# **edit** dynamic-profiles ***profile-name***

2. Specify that you want to configure the interfaces for the dynamic profile.

   [edit dynamic-profiles *profile-name*]
   user@host# **edit** interfaces

3. Configure the dynamic ACI interface set.

   [edit dynamic-profiles *profile-name* interfaces]
   user@host# **edit** interface-set **$junos-interface-set-name**

   You must use the **$junos-interface-set-name** predefined dynamic variable to represent the name of the ACI interface set. The **$junos-interface-set-name** variable is dynamically replaced with the actual ACI interface set name generated by the router when the first DHCP or PPPoE subscriber from that household logs in.

4. Include the interfaces for the dynamic ACI interface set.

   [edit dynamic-profiles *profile-name* interfaces interface-set
     “$junos-interface-set-name”]
   user@host# **set** interface **$junos-interface-ifd-name**

   You must use the **$junos-interface-ifd-name** predefined dynamic variable to represent the name of the ACI interface set. The **$junos-interface-ifd-name** variable is dynamically replaced with the name of the interface on which the DHCP or PPPoE subscriber accesses the router.

NOTE: The unit *logical-unit-number* statement is not required in the dynamic profile at the [edit dynamic-profiles *profile-name* interfaces interface-set *interface-set-name* interface *interface-name*] hierarchy level when you configure an ACI interface set.

5. (Optional) For dynamic PPPoE subscriber interfaces, configure the maximum number of dynamic PPPoE sessions that the router can activate for the ACI interface set.

   ```
   [edit dynamic-profiles profile-name interfaces interface-set
     "$junos-interface-set-name"]
   user@host# edit pppoe-underlying-options
   [edit dynamic-profiles profile-name interfaces interface-set "$junos-interface-set-name"
     pppoe-underlying-options]
   user@host# set max-sessions number
   ```

   Issuing the **max-sessions** statement in a dynamic profile for an ACI interface set limits the maximum number of dynamic PPPoE sessions at the ACI interface set level from the same household.

6. (Optional) Apply attributes for CoS and interface filters to all subscriber interfaces belonging to the ACI interface set.

The following example shows the minimum dynamic profile required to define an ACI interface set named aci-vlan-set-profile. The **interface-set** stanza uses predefined dynamic variables to represent the interface set (**$junos-interface-set-name**) and the underlying physical interface (**$junos-interface-ifd-name**).

```
[edit dynamic-profiles aci-vlan-set-profile]
interfaces {
  interface-set "$junos-interface-set-name" {
    interface "$junos-interface-ifd-name";
  }
}
```

The following example shows a more complex dynamic profile for an ACI interface set named aci-vlan-set-profile-pppoe-cos. In addition to the required **interface-set** stanza, this profile includes optional attributes for PPPoE (**max-sessions** statement) and CoS. The router applies these PPPoE and CoS attributes to all subscriber interfaces from the same household, which is represented by the ACI interface set.

```
[edit dynamic-profiles aci-vlan-set-profile-pppoe-cos]
variables {
  ds1q1q2DP uid;
  ef1_dp uid;
}
  interfaces {
    interface-set "$junos-interface-set-name" {
      interface "$junos-interface-ifd-name";
      pppoe-underlying-options {
        max-sessions 3;
      }
    }
  }
```



```
class-of-service {
  traffic-control-profiles {
    tcp2 {
      scheduler-map "$ds1q1q2DP";
      shaping-rate 50m;
      overhead-accounting bytes -20;
      guaranteed-rate 30m;
    }
  }
  interfaces {
    interface-set "$junos-interface-set-name" {
      output-traffic-control-profile tcp2;
    }
  }
  scheduler-maps {
    "$ds1q1q2DP" {
      forwarding-class ef scheduler "$ef1_dp";
    }
  }
  schedulers {
    "$ef1_dp" {
      transmit-rate percent 25;
      priority low;
    }
  }
}
```

Related Documentation

- Agent Circuit Identifier-Based Dynamic VLANs Overview on page 25
- Agent Circuit Identifier-Based Dynamic VLANs Components Overview on page 27
- Configuring Dynamic VLANs Based on Agent Circuit Identifier Information on page 29
- Verifying and Managing Agent Circuit Identifier-Based Dynamic VLAN Configuration on page 38
- Clearing Agent Circuit Identifier Interface Sets on page 39

- *Applying CoS Attributes to VLANs Using Agent-Circuit-Identifiers*
- *Example: Implementing a Filter for Households That Use ACI-Based VLANs*

Configuring Dynamic Underlying VLAN Interfaces to Use Agent Circuit Identifier Information

After you define the agent circuit identifier (ACI) interface set, you must configure the underlying VLAN interface to enable creation of dynamic VLAN subscriber interfaces based on ACI information. You can configure the underlying VLAN interface statically or dynamically.

This topic describes how to configure the underlying VLAN interface *dynamically*.

Before you begin:

- Create a dynamic profile that defines the underlying VLAN interface.

See the following topics:

- *Configuring a Basic Dynamic Profile*
- *Configuring VLAN Dynamic Profiles*
- *Configuring VLAN Interfaces to Use Dynamic Profiles*

To configure a dynamic underlying VLAN interface to use ACI information:

- In the dynamic profile for the underlying VLAN interface, associate the dynamic profile that defines the ACI interface set with the underlying VLAN interface.

  ```
  [edit dynamic-profiles profile-name]
  user@host# set interfaces interface-name unit logical-unit-number auto-configure
    agent-circuit-identifier dynamic-profile aci-interface-set-profile-name
  ```

  For example, the following statement in a dynamic profile named aci-vlan-underlying-profile-demux associates the dynamic underlying VLAN interface with dynamic profile aci-vlan-set-profile2 that defines the ACI interface set. You must use the predefined dynamic variable **$junos-interface-ifd-name** to represent the interface name, and **$junos-interface-unit** to represent the logical unit number.

  ```
  [edit dynamic-profiles aci-vlan-underlying-profile-demux]
  user@host# set interfaces “$junos-interface-ifd-name” unit “$junos-interface-unit”
    auto-configure agent-circuit-identifier dynamic-profile aci-vlan-set-profile2
  ```

The following example shows the dynamic configuration that uses this statement. This configuration enables the underlying dynamic IP demultiplexing (IP demux) VLAN interface to create dynamic subscriber interfaces based on ACI information by applying a single default ACI interface set dynamic profile (aci-vlan-set-profile2) to all households on the VLAN interface.

```
[edit dynamic-profiles aci-vlan-underlying-profile-demux]
interfaces {
  “$junos-interface-ifd-name” {
    unit "$junos-interface-unit" {
      auto-configure {
        agent-circuit-identifier {
          dynamic-profile aci-vlan-set-profile2;
        }
      }
      vlan-id "$junos-vlan-id";
      demux-options {
        underlying-interface “$junos-interface-ifd-name”;
      }
      family inet {
        unnumbered-address lo0.0 preferred-source-address 100.20.0.2;
      }
    }
  }
}
```

Related Documentation

- Agent Circuit Identifier-Based Dynamic VLANs Overview on page 25
- Agent Circuit Identifier-Based Dynamic VLANs Components Overview on page 27

- Configuring Dynamic VLANs Based on Agent Circuit Identifier Information on page 29
- Verifying and Managing Agent Circuit Identifier-Based Dynamic VLAN Configuration on page 38

Configuring Static Underlying VLAN Interfaces to Use Agent Circuit Identifier Information

After you define the agent circuit identifier (ACI) interface set, you must configure the underlying VLAN interface to enable creation of dynamic VLAN subscriber interfaces based on ACI information. You can configure the underlying VLAN interface statically or dynamically.

This topic describes how to configure the underlying VLAN interface *statically*.

To configure a static underlying VLAN interface to use ACI information:

- Associate the dynamic profile that defines the ACI interface set with the static underlying VLAN interface.

  [edit]
  user@host# **set interfaces *interface-name* unit *logical-unit-number* auto-configure agent-circuit-identifier dynamic-profile *aci-interface-set-profile-name***

  For example, the following statement associates static Gigabit Ethernet VLAN interface ge-1/0/0.0 with the dynamic profile aci-vlan-set-profile that defines the ACI interface set.

  [edit]
  user@host# **set interfaces ge-1/0/0 unit 0 auto-configure agent-circuit-identifier dynamic-profile aci-vlan-set-profile**

The following example shows the static configuration that uses this statement. This configuration enables the underlying VLAN interface ge-1/0/0.0 to create dynamic subscriber interfaces based on ACI information by applying a single default ACI interface set dynamic profile (aci-vlan-set-profile) to all households on the VLAN interface.

```
[edit]
interfaces {
  ge-1/0/0 {
    flexible-vlan-tagging;
    unit 0 {
      vlan-id 100;
      auto-configure {
        agent-circuit-identifier {
          dynamic-profile aci-vlan-set-profile;
        }
      }
    }
  }
}
```

Related Documentation

- Agent Circuit Identifier-Based Dynamic VLANs Overview on page 25

- Agent Circuit Identifier-Based Dynamic VLANs Components Overview on page 27
- Configuring Dynamic VLANs Based on Agent Circuit Identifier Information on page 29
- Verifying and Managing Agent Circuit Identifier-Based Dynamic VLAN Configuration on page 38

Configuring Dynamic VLAN Subscriber Interfaces Based on Agent Circuit Identifier Information

After you define the dynamic agent circuit identifier (ACI) interface set and enable creation of ACI-based dynamic VLAN subscriber interfaces on the underlying VLAN interface, you must complete the configuration by associating the ACI interface set with the PPPoE or IP demultiplexing (IP demux) subscriber interface in the dynamic profile for the subscriber interface.

Before you begin:

- Create a dynamic profile that defines the logical subscriber interface.

  See the following topics:

  - *Configuring a Basic Dynamic Profile*
  - Configuring Dynamic PPPoE Subscriber Interfaces Using Dynamic Profiles on page 145
  - Configuring Dynamic Subscriber Interfaces Using IP Demux Interfaces in Dynamic Profiles on page 77

To configure a dynamic VLAN subscriber interface based on ACI information:

- In the dynamic profile for the PPPoE or IP demux subscriber interface, associate the dynamic ACI interface set with the dynamic VLAN subscriber interface name (**pp0** or **demux0**) and logical unit number.

  [edit dynamic-profiles *profile-name*]
  user@host# **set** interfaces interface-set **$junos-interface-set-name** interface *interface-name* unit **$junos-interface-unit**

  For example, the following statement in a dynamic profile named aci-vlan-pppoe-profile associates the dynamic ACI interface set with the dynamic **pp0** (PPPoE) logical subscriber interface. You must use the predefined dynamic variable **$junos-interface-set-name** to represent the name of the dynamic ACI interface set, and **$junos-interface-unit** to represent the logical unit number of the subscriber interface.

  [edit dynamic-profiles aci-vlan-pppoe-profile]
  user@host# **set interfaces interface-set $junos-interface-set-name interface pp0 unit $junos-interface-unit**

  Similarly, the following statement in a dynamic profile named aci-vlan-demux-profile associates the dynamic ACI interface set (represented by **$junos-interface-set-name**) with the **demux0** (IP demux) logical subscriber interface.

  [edit dynamic-profiles aci-vlan-demux-profile]


```
user@host# set interfaces interface-set $junos-interface-set-name interface demux0
  unit $junos-interface-unit
```

The following examples show the dynamic configurations that use each of these statements. The following sample configuration shows a dynamic profile named aci-vlan-pppoe-profile for an ACI-based dynamic PPPoE (**pp0**) subscriber interface for use by PPPoE subscribers.

```
[edit dynamic-profiles aci-vlan-pppoe-profile]
interfaces {
  interface-set "$junos-interface-set-name" {
    interface pp0 {
      unit "$junos-interface-unit";
    }
  }
  pp0 {
    unit "$junos-interface-unit" {
      ppp-options {
        chap;
        pap;
      }
      pppoe-options {
        underlying-interface "$junos-underlying-interface";
        server;
      }
      no-keepalives;
      family inet {
        unnumbered-address lo0.0;
      }
    }
  }
}
```

The following sample configuration shows a dynamic profile named aci-vlan-demux-profile for an ACI-based dynamic IP demux(**demux0**) subscriber interface for use by DHCP subscribers.

```
[edit dynamic-profiles aci-vlan-demux-profile]
interfaces {
  interface-set "$junos-interface-set-name" {
    interface demux0 {
      unit "$junos-interface-unit";
    }
  }
  demux0 {
    unit "$junos-interface-unit" {
      demux-options {
        underlying-interface "$junos-underlying-interface";
      }
      family inet {
        demux-source {
          $junos-subscriber-ip-address;
        }
        unnumbered-address lo0.0 preferred-source-address 100.20.200.202;
      }
    }
```

```
    }
  }
```

Related Documentation

- Agent Circuit Identifier-Based Dynamic VLANs Overview on page 25
- Agent Circuit Identifier-Based Dynamic VLANs Components Overview on page 27
- Configuring Dynamic VLANs Based on Agent Circuit Identifier Information on page 29
- Verifying and Managing Agent Circuit Identifier-Based Dynamic VLAN Configuration on page 38
- Clearing Agent Circuit Identifier Interface Sets on page 39

Verifying and Managing Agent Circuit Identifier-Based Dynamic VLAN Configuration

Purpose View information about dynamic agent circuit identifier (ACI) interface sets and ACI-based dynamic VLAN subscriber interfaces configured on the router.

Action

- To display the logical and physical interface associations for the classifier, rewrite rules, scheduler map objects, and CoS adjustment settings:

  `user@host>` **show class-of-service interface** ***interface-name***

- To display the CoS associations for the specified dynamic ACI interface set:

  `user@host>` **show class-of-service interface-set** ***aci-interface-set-name***

- To display information about the specified CoS traffic shaping and scheduling profile:

  `user@host>` **show class-of-service traffic-control-profile** ***profile-name***

- To display address bindings and ACI interface set information in the client table on the extended DHCP local server:

  `user@host>` **show dhcp server binding detail**

- To display status information about a specified Gigabit Ethernet interface:

  `user@host>` **show interfaces ge-*fpc/pic/port.logical-unit-number***

- To display status information about a specified IP demultiplexing (IP demux) interface:

  `user@host>` **show interfaces demux0.*logical-interface-number***

- To display information about all dynamic ACI interface sets configured on the router:

  `user@host>` **show interfaces interface-set**

- To display session-specific information about ACI-based dynamic PPPoE subscriber interfaces:

  `user@host>` **show pppoe interfaces pp0.*logical-unit-number***

- To display information about PPPoE underlying interfaces, including whether creation of ACI-based dynamic VLAN subscriber interfaces is enabled on the underlying interface:

  `user@host>` **show pppoe underlying-interfaces *logical-interface-name* detail**

- To display information about active subscriber sessions associated with ACI interface sets:

  `user@host>` **show subscribers detail**

- To display information about active subscriber sessions associated with a specified ACI interface set:

`user@host>` show subscribers **aci-interface-set-name** ***aci-interface-set-name*** **detail**

- To display information about active subscriber sessions that have an agent circuit identifier value containing a matching substring:

  `user@host>` show subscribers **agent-circuit-identifier** ***agent-circuit-identifier-substring*** **detail**

Related Documentation

- Agent Circuit Identifier-Based Dynamic VLANs Overview on page 25
- Agent Circuit Identifier-Based Dynamic VLANs Components Overview on page 27
- Configuring Dynamic VLANs Based on Agent Circuit Identifier Information on page 29
- Clearing Agent Circuit Identifier Interface Sets on page 39
- CLI Explorer

Clearing Agent Circuit Identifier Interface Sets

Purpose Clear a specified dynamic agent circuit identifier (ACI) interface set configured on the router.

Action

- To clear a specified ACI interface set that that has no active members:

  `user@host>` clear auto-configuration interfaces interface-set ***interface-set-name***

  For example, the following command clears the ACI interface set named aci-1003-ge-1/0/0.4001:

```
user@host> clear auto-configuration interfaces interface-set aci-1003-ge-1/0/0.4001
Interface-set aci-1003-ge-1/0/0.4001 deleted
```

Meaning The router dynamically creates an ACI interface set, if configured, when the first DHCP or PPPoE subscriber from a particular household logs in. However, the router does not automatically delete the ACI interface set when the last subscriber from that household logs out. As a result, you must use the **clear auto-configuration interfaces interface-set** command to explicitly clear the ACI interface set when it no longer has any active subscriber interface members. If you attempt to clear an ACI interface that still has active member interfaces, the router displays an error message and rejects the command.

When you specify the name of the ACI interface set to be cleared, you must use the ACI interface set name internally generated by the router, and not the actual ACI string carried in DHCP and PPPoE control packets. The router uses the following format to name ACI interface sets, as shown in the ACI interface set named aci-1003-ge-1/0/0.4001:

aci-*nnnn-interface-name.logical-unit-number*

where:

- *nnnn* is a randomly generated 4-digit identifier (1003 in the example)
- *interface-name* is the name of the dynamic subscriber interface (ge-1/0/0 in the example)
- *logical-unit-number* is the logical unit number of the dynamic subscriber interface (4001 in the example)

To view the names of the ACI interface sets configured on the router, use the **show subscribers** command.

Related Documentation

- Agent Circuit Identifier-Based Dynamic VLANs Components Overview on page 27
- Configuring Dynamic VLANs Based on Agent Circuit Identifier Information on page 29
- Verifying and Managing Agent Circuit Identifier-Based Dynamic VLAN Configuration on page 38
- CLI Explorer

CHAPTER 5

Using VXLANs in a Subscriber Management Network

Understanding VXLANs

Virtual eXtensible LAN protocol (VXLAN) technology allows networks to support more VLANs. According to the IEEE 802.1Q standard, traditional VLAN identifiers are 12-bits long—this naming limits networks to 4094 VLANs. The VXLAN protocol overcomes this limitation by using a longer logical network identifier that allows more VLANs and, therefore, more logical network isolation for large networks such as clouds that typically include many virtual machines.

VXLAN Benefits

VXLAN technology allows you to segment your networks (as VLANs do) but it provides benefits that VLANs cannot. Here are the most important benefits of using VXLANs:

- You can theoretically create as many as 16 million VXLANs in an administrative domain (as opposed to 4094 VLANs on a Juniper Networks device).

  - MX Series routers and EX9200 switches support as many as 32K VXLANs, 32K multicast groups, and 8K virtual tunnel endpoints (VTEPs). This means that VXLANs based on MX Series routers provide network segmentation at the scale required by cloud builders to support very large numbers of tenants.
  - QFX 5100 switches support 4K VXLANs, 4K multicast groups, and 2K VTEPs.
- You can enable migration of virtual machines between servers that exist in separate Layer 2 domains by tunneling the traffic over Layer 3 networks. This functionality allows you to dynamically allocate resources within or between data centers without being constrained by Layer 2 boundaries or being forced to create large or geographically stretched Layer 2 domains.

Using VXLANs to create smaller Layer 2 domains that are connected over a Layer 3 network means that you don’t need to use STP to converge the topology but can use more-robust routing protocols in the Layer 3 network instead. In the absence of STP, none of your links are blocked, which means you can get full value from all the ports that you purchase. Using routing protocols to connect your Layer 2 domains also allows you to load balance the traffic to ensure that you get the best use of your available bandwidth. Given the amount of east-west traffic that often flows within or between data centers, maximizing your network performance for that traffic is very important.

The video *Why Use an Overlay Network in a Data Center?* presents a brief overview of the advantages of using VXLANs.

Video: Why Use an Overlay Network in a Data Center?

How Does VXLAN Work?

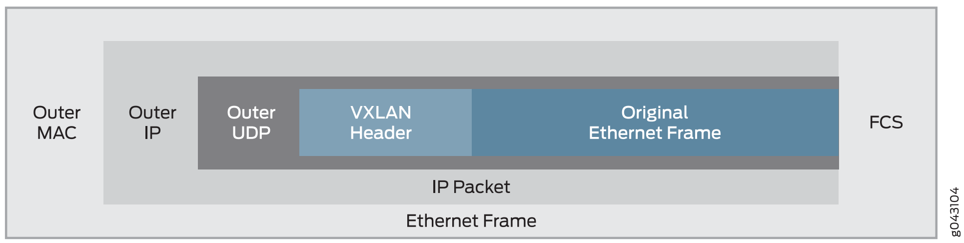
VXLAN is often described as an overlay technology because it allows you to stretch Layer 2 connections over an intervening Layer 3 network by encapsulating (tunneling) Ethernet frames in a VXLAN packet that includes IP addresses. Devices that support VXLANs are called virtual tunnel endpoints (VTEPs)—they can be end hosts or network switches or routers. VTEPs encapsulate VXLAN traffic and de-encapsulate that traffic when it leaves the VXLAN tunnel. To encapsulate an Ethernet frame, VTEPs add a number of fields, including the following:

- Outer MAC destination address (MAC address of the tunnel endpoint VTEP)
- Outer MAC source address (MAC address of the tunnel source VTEP)
- Outer IP destination address (IP address of the tunnel endpoint VTEP)
- Outer IP source address (IP address of the tunnel source VTEP)
- Outer UDP header
- A VXLAN header that includes a 24-bit field—called the VXLAN network identifier (VNI)—that is used to uniquely identify the VXLAN. The VNI is similar to a VLAN ID, but having 24 bits allows you to create many more VXLANs than VLANs.

NOTE: Because VXLAN adds 50 to 54 bytes of additional header information to the original Ethernet frame, you might want to increase the MTU of the underlying network. In this case, configure the MTU of the physical interfaces that participate in the VXLAN network, not the MTU of the logical VTEP source interface, which is ignored.

Figure 2 on page 43 shows the VXLAN packet format.

Figure 2: VXLAN Packet Format



VXLAN Implementation Methods

Junos OS supports implementing VXLANs in the following environments:

- Without SDN controllers. VXLANs implemented in this type of environment are known as manual VXLANs.
- With SDN controllers. In this environment, SDN controllers use the Open vSwitch Database (OVSDB) management protocol to provide a means through which controllers (such as a VMware NSX or Juniper Contrail controller) and Juniper Networks devices that support OVSDB can communicate.

Using a QFX5100 Switch with VXLANs

You can configure a QFX5100 switch to perform all of the following roles:

- In an environment without an SDN controller, act as a transit Layer 3 switch for downstream hosts acting as VTEPs. In this configuration, you do not need to configure any VXLAN functionality on the switch. You do need to configure IGMP and PIM so that the switch can form the multicast trees for the VXLAN multicast groups. (See Manual VXLANs Require PIM on page 45 for more information.)
- In an environment with or without an SDN controller, act as a Layer 2 gateway between virtualized and non-virtualized networks in the same data center or between data centers. For example, you can use a QFX5100 switch to connect a network that uses VXLANs to one that uses VLANs.
- Act as a Layer 2 gateway between virtualized networks in the same or different data centers and allow virtual machines to move (VMotion) between those networks and data centers. For example, if you want to allow VMotion between devices in two different networks, you can create the same VLAN in both networks and put both

devices on that VLAN. The QFX5100 switches connected to these devices, acting as VTEPs, can map that VLAN to the same VXLAN, and the VXLAN traffic can then be routed between the two networks.

NOTE: A QFX 5100 switch cannot route traffic between different VXLANs. To connect devices in different VXLANs you need a VXLAN-capable Layer 3 gateway, such as a Juniper Networks MX Series router.

Because the additional headers add 50 to 54 bytes, you might need to increase the MTU on a QFX5100 VTEP to accommodate larger packets. For example, if the switch is using the default MTU value of 1514 bytes and you want to forward 1500-byte packets over the VXLAN, you need to increase the MTU to allow for the increased packet size caused by the additional headers.

Changing the UDP Port on a QFX5100 Switch

Starting with Junos OS 14.1X53-D25, you can configure the UDP port used as the destination port for VXLAN traffic on a QFX5100 switch. To configure the VXLAN destination port to be something other than the default UDP port of 4789, enter

set protocols l2-learning destination-udp-port *port-number*

The port you configure will be used for all VXLANs configured on the switch.

NOTE: If you make this change on one switch in a VXLAN, you must make the same change on all the devices that terminate the VXLANs configured on your switch. If you do not do so, traffic will be disrupted for all the VXLANs configured on your switch. When you change the UDP port, the previously learned remote VTEPs and remote MACs are lost and VXLAN traffic is disrupted until the switch relearns the remote VTEPs and remote MACs.

Controlling Transit Multicast Traffic on a QFX5100 Switch

When a QFX5100 switch acting as a VXLAN VTEP receives a broadcast, unknown unicast, or multicast packet, it performs the following actions on the packet:

1. It de-encapsulates the packet and delivers it to the locally attached hosts.
2. It then adds the VXLAN encapsulation again and sends the packet to the other VTEPs in the VXLAN.

These actions are performed by the loopback interface used as the VXLAN tunnel address and can therefore negatively impact the bandwidth available to the VTEP. With Junos OS 14.1X53-D30 and later, if you know that there are no multicast receivers attached to other VTEPs in the VXLAN who want traffic for a specific multicast group, you can reduce the processing load on the loopback interface by entering the following statement:

set protocols l2-learning disable-vxlan-multicast-transit vxlan-multicast-group *multicast-group*

In this case, no traffic will be forwarded for the specified group but all other multicast traffic will be forwarded. If you do not want to forward any multicast traffic to other VTEPs in the VXLAN, enter the following statement:

set protocols l2-learning disable-vxlan-multicast-transit vxlan-multicast-group all

Using an MX Series Router or EX9200 Switch as a VTEP

You can configure an MX Series router or EX9200 switch to act as a VTEP and perform all of the following roles:

- Act as a Layer 2 gateway between virtualized and non-virtualized networks in the same data center or between data centers. For example, you can use an MX Series router to connect a network that uses VXLANs to one that uses VLANs.
- Act as a Layer 2 gateway between virtualized networks in the same or different data centers and allow virtual machines to move (VMotion) between those networks and data centers.
- Act as a Layer 3 gateway to route traffic between different VXLANs in the same data center.
- Act as a Layer 3 gateway to route traffic between different VXLANs in different data centers over a WAN or the Internet using standard routing protocols or VPLS tunnels.

NOTE: **If you want an MX Series router or EX9200 switch to be a VXLAN Layer 3 gateway, you must configure integrated routing and bridging (IRB) interfaces to connect the VXLANs, just as you do if you want to route traffic between VLANs.**

Manual VXLANs Require PIM

In an environment with a controller (such as VMware's NSX), you can provision VXLANs on a Juniper Networks device. A controller also provides a control plane that VTEPs use to advertise their reachability and learn about the reachability of other VTEPs. You can also manually create VXLANs on Juniper Networks devices instead of using a controller. If you use this approach, you must also configure PIM on the VTEPs so that they can create VXLAN tunnels between themselves.

You must also configure each VTEP in a given VXLAN to be a member of the same multicast group. (If possible, you should assign a different multicast group address to each VXLAN, though this is not required. Multiple VXLANs can share the same multicast group.) The VTEPs can then forward ARP requests they receive from their connected hosts to the multicast group. The other VTEPs in the group de-encapsulate the VXLAN information, and (assuming they are members of the same VXLAN) they forward the ARP request to their connected hosts. When the target host receives the ARP request, it responds with its MAC address, and its VTEP forwards this ARP reply back to the source VTEP. Through this process, the VTEPs learn the IP addresses of the other VTEPs in the VXLAN and the MAC addresses of the hosts connected to the other VTEPs.

The multicast groups and trees are also used to forward broadcast, unknown unicast, and multicast (BUM) traffic between VTEPs. This prevents BUM traffic from being unnecessarily flooded outside the VXLAN.

NOTE: Multicast traffic that is forwarded through a VXLAN tunnel is sent only to the remote VTEPs in the VXLAN. That is, the encapsulating VTEP does not copy and send copies of the packets according to the multicast tree—it only forwards the received multicast packets to the remote VTEPs. The remote VTEPs de-encapsulate the encapsulated multicast packets and forward them the appropriate Layer 2 interfaces. The remote VTEPs also do not copy and send copies of the packets according to the multicast tree.

Load Balancing VXLAN Traffic

On QFX5100 switches, the Layer 3 routes that form VXLAN tunnels use per-packet load balancing by default, which means that load balancing is implemented if there are ECMP paths to the remote VTEP. This is different from normal routing behavior in which per-packet load balancing is not used by default. (Normal routing uses per-prefix load balancing by default.)

The source port field in the UDP header is used to enable ECMP load balancing of the VXLAN traffic in the Layer 3 network. This field is set to a hash of the inner packet fields, which results in a variable that ECMP can use to distinguish between tunnels (flows). (None of the other fields that flow-based ECMP normally uses are suitable for use with VXLANs. All tunnels between the same two VTEPs have the same outer source and destination IP addresses, and the UDP destination port is set to port 4789 by definition. Therefore, none of these fields provide a sufficient way for ECMP to differentiate flows.)

Using ping and traceroute With a VXLAN

On a QFX5100 switch, you can use the **ping** and **traceroute** commands to troubleshoot traffic flow through a VXLAN tunnel by including the **overlay** parameter and various options. You use these options to force the **ping** or **traceroute** packets to follow the same path as data packets through the VXLAN tunnel. In other words, you make the underlay packets (**ping** and **traceroute**) take the same route as the overlay packets (data traffic). See *ping overlay* and *traceroute overlay* for more information.

VXLAN Constraints on QFX5100 Switches

When configuring VXLANs on QFX5100 switches, be aware of the constraints in the following list. In this list, “Layer 3 side” refers to a network-facing interface that performs VXLAN encapsulation and de-encapsulation, and “Layer 2 side” refers to a server-facing interface that is a member of a VLAN that is mapped to a VXLAN.

- You can use VXLANs on a Virtual Chassis or Virtual Chassis Fabric if all of the members are QFX5100 switches. You cannot use VXLANs if any of the members is not a QFX5100 switch.
- VXLAN configuration is supported only in the default routing instance.

- A QFX5100 switch cannot route traffic between different VXLANs.
- A physical interface cannot be a member of a VLAN and a VXLAN. That is, an interface that performs VXLAN encapsulation and de-encapsulation cannot also be a member of a VLAN. For example, if a VLAN that is a mapped to a VXLAN is a member of trunk port xe-0/0/0, any other VLAN that is a member of xe-0/0/0 must also be assigned to a VXLAN.
- Multichassis link aggregation groups (MC-LAGS) are not supported with VXLAN.
- IP fragmentation and defragmentation are not supported on the Layer 3 side.
- The following features are not supported on the Layer 2 side:
  - STP (any variant)
  - IGMP snooping
- Access port security features are not supported with VXLAN. For example, the following features are not supported:
  - DHCP snooping
  - dynamic ARP inspection
  - MAC limiting and MAC move limiting
- See the following to determine whether ingress node replication is supported on QFX switches:
  - PIM used for control plane: ingress node replication is not supported.
  - Control plane provided by a controller: ingress node replication is not supported.
  - EVPN used with VXLANs: ingress node replication is supported.
- PIM-BIDIR and PIM-SSM are not supported with VXLANs.
- Class of service (CoS) features are not supported with VXLANs.
- If you configure a port-mirroring instance to mirror traffic egressing from an interface that performs VXLAN encapsulation, the source and destination MAC addresses of the mirrored packets are invalid. The original VXLAN traffic is not affected.

Related Documentation

- *Examples: Manually Configuring VXLANs on QFX Series Switches*
- Example: Manually Configuring VXLANs on MX Series Routers on page 47
- *OVSDB Support on Juniper Networks Devices*
- *mtu*

Example: Manually Configuring VXLANs on MX Series Routers

Virtual Extensible Local Area Network (VXLAN) is a Layer 3 encapsulation protocol that enables MX Series routers to push Layer 2 or Layer 3 packets through a VXLAN tunnel to a virtualized data center or the Internet. Communication is established between two virtual tunnel endpoints (VTEPs). VTEPs encapsulate the virtual machine traffic into a VXLAN header and strip off the encapsulation.

This example shows how to configure VXLAN on MX Series routers using switch options in a default bridge domain.

Requirements

This example uses the following hardware and software components:

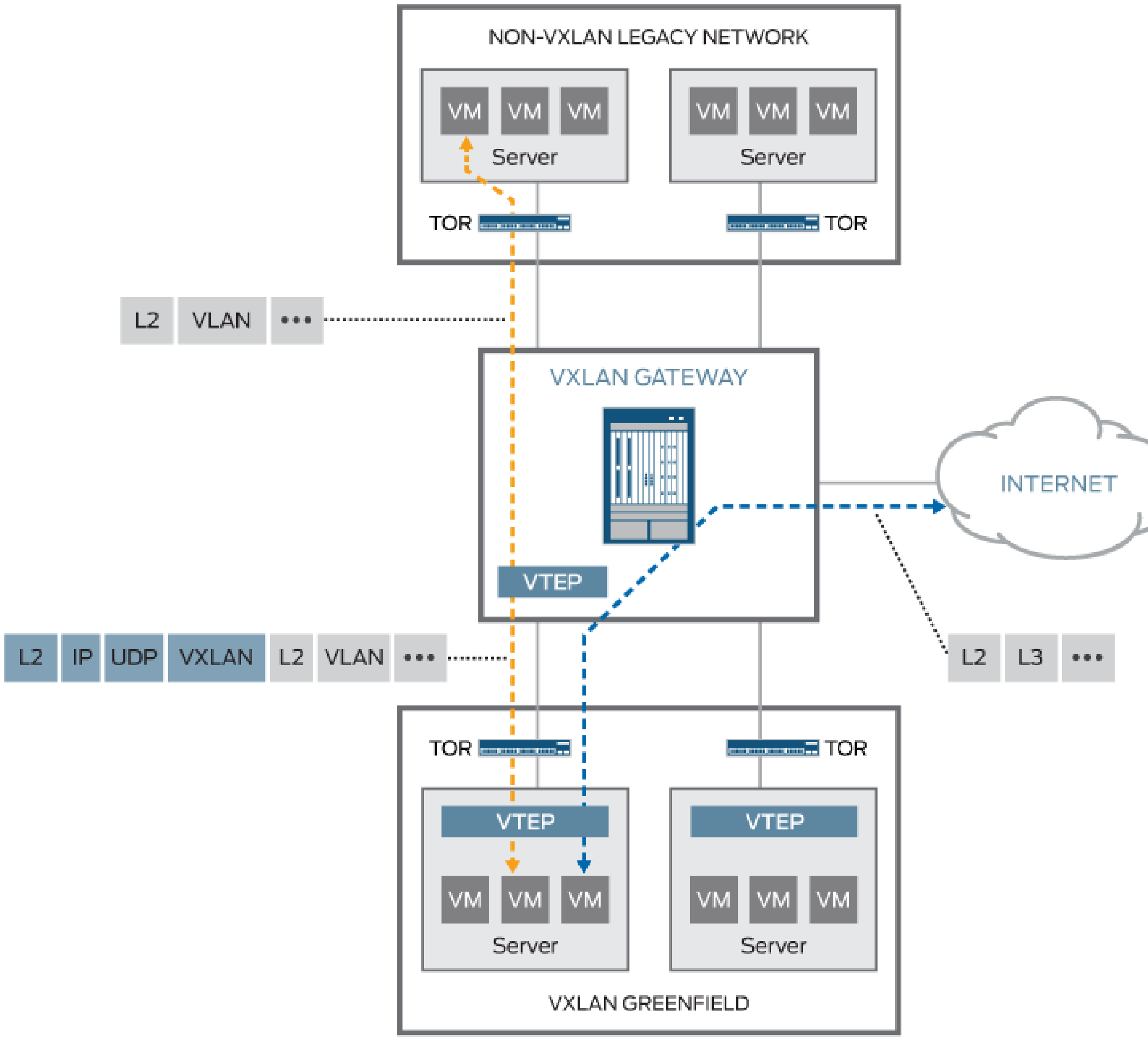
- An MX Series router
- A VXLAN capable peer router
- Junos OS Release 14.1

Overview

In this example, VXLAN is configured to run on a default bridge domain. VTEP interfaces sources are configured to the loopback address, and VLAN groups are configured under bridge domains with VXLAN enabled. Interfaces are configured for VLAN tagging and encapsulation, and IRB is enabled. OSPF and PIM protocols are configured to facilitate unicast and multicast routing. The chassis is configured for GRES and enhanced IP services.

Topology

Figure 1: VXLAN Topology



Configuring VXLAN on MX Series Routers

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 switch-options vtep-source-interface lo0.0
set bridge-domains vlan-5 vxlan vni 100
set bridge-domains vlan-5 vxlan multicast-group 239.1.1.1
set bridge-domains vlan-5 vlan-id 100
set bridge-domains vlan-5 routing-interface irb.0
set bridge-domains vlan-5 interface xe-1/0/0.0
```

set bridge-domains vlan-6 vxlan vni 200
set bridge-domains vlan-6 vxlan multicast-group 239.1.1.1
set bridge-domains vlan-6 vlan-id 200
set bridge-domains vlan-6 routing-interface irb.1
set bridge-domains vlan-6 interface xe-2/0/0.0
set interfaces xe-1/0/0 vlan-tagging
set interfaces xe-1/0/0 encapsulation flexible-ethernet-services
set interfaces xe-1/0/0 unit 0 encapsulation vlan-bridge
set interfaces xe-1/0/0 unit 0 vlan-id 100
set interfaces xe-2/0/0 vlan-tagging
set interfaces xe-2/0/0 encapsulation flexible-ethernet-services
set interfaces xe-2/0/0 unit 0 encapsulation vlan-bridge
set interfaces xe-2/0/0 unit 0 vlan-id 200
set interface irb unit 0 family inet address 5.5.5.1/24
set interface irb unit 1 family inet address 6.6.6.1/24
set interfaces lo0 unit 0 family inet address 3.3.3.3/32
set protocols ospf area 0.0.0.0 interface ge-8/3/8.0
set protocols ospf area 0.0.0.0 interface lo0.0
set protocols ospf area 0.0.0.0 interface xe-0/1/3.0
set protocols ospf area 0.0.0.0 interface ge-8/3/2.0
set protocols pim rp static address 10.2.1.3
set protocols pim interface lo0.0 mode bidirectional-sparse
set protocols pim interface ge-8/3/8.0 mode bidirectional-sparse
set protocols pim interface xe-0/1/3.0 mode bidirectional-sparse
set protocols pim interface ge-8/3/2.0 mode bidirectional-sparse
set chassis redundancy graceful-switchover
set chassis aggregated-devices ethernet device-count 10
set chassis fpc 1 pic 0 tunnel-services bandwidth 10g
set chassis network-services enhanced-ip

Configuring VXLAN

Step-by-Step Procedure

The following example show how to set up a basic VXLAN configuration with default bridge domains and switch options. To configure VXLAN on an MX Series router, follow these steps:

1. Configure VTEP interface sources under **switch-options** for the default-switch.

```
[edit]
user@router# set switch-options vtep-source-interface lo0.0
```

2. Set up a VLAN group named **vlan-5** and set its VXLAN Network Identifier (VNI) to 100.

```
[edit]
user@router# set bridge-domains vlan-5 vxlan vni 100
```

3. Configure the **vlan-5** multicast group address for VXLAN.

```
[edit]
user@router# set bridge-domains vlan-5 vxlan multicast-group 239.1.1.1
```

4. Set the VLAN ID to 100 for **vlan-5**.

```
[edit]
user@router# set bridge-domains vlan-5 vlan-id 100
```


5. Configure integrated bridging and routing (IRB) for **vlan-5**.

```
[edit]
user@router# set bridge-domains vlan-5 routing-interface irb.0
```

6. Assign the xe-1/0/0.0 interface to **vlan-5**.

```
[edit]
user@router# set bridge-domains vlan-5 interface xe-1/0/0.0
```

7. Set up a VLAN group named **vlan-6** and set its VXLAN Network Identifier (VNI) to 200.

```
[edit]
user@router# set bridge-domains vlan-6 vxlan vni 200
```

8. Configure the **vlan-6** multicast group address for VXLAN.

```
[edit]
user@router# set bridge-domains vlan-6 vxlan multicast-group 239.1.1.1
```

9. Set the VLAN ID to 100 for **vlan-6**.

```
[edit]
user@router# set bridge-domains vlan-6 vlan-id 200
```

10. Configure IRB for **vlan-6**.

```
[edit]
user@router# set bridge-domains vlan-6 routing-interface irb.1
```

11. Assign the xe-2/0/0.0 interface to **vlan-6**.

```
[edit]
user@router# set bridge-domains vlan-6 interface xe-2/0/0.0
```

12. Set up VLAN tagging for xe-1/0/0.

```
[edit]
user@router# set interfaces xe-1/0/0 vlan-tagging
```

13. Configure flexible Ethernet service encapsulation on xe-1/0/0.

```
[edit]
user@router# set interfaces xe-1/0/0 encapsulation flexible-ethernet-services
```

14. Set up VLAN bridging encapsulation for xe-1/0/0 unit 0 .

```
[edit]
user@router# set interfaces xe-1/0/0 unit 0 encapsulation vlan-bridge
```

15. Set the xe-1/0/0 unit 0 VLAN ID to 100.

```
[edit]
user@router# set interfaces xe-1/0/0 unit 0 vlan-id 100
```

16. Configure VLAN tagging for xe-2/0/0

```
[edit]
user@router# set interfaces xe-2/0/0 vlan-tagging
```

17. Set up flexible Ethernet service encapsulation on xe-2/0/0.

```
[edit]
user@router# set interfaces xe-2/0/0 encapsulation flexible-ethernet-services
```

18. Configure VLAN bridging encapsulation for xe-2/0/0 unit 0˙.

```
[edit]
user@router# set interfaces xe-2/0/0 unit 0 encapsulation vlan-bridge
```

19. Set the xe-2/0/0 unit 0 VLAN ID to 200.

```
[edit]
user@router# set interfaces xe-2/0/0 unit 0 vlan-id 200
```

20. Configure the IRB unit 0 family inet address.

```
[edit]
user@router# set interface irb unit 0 family inet address 5.5.5.1/24
```

21. Configure the IRB unit 1 family inet address.

```
[edit]
user@router# set interface irb unit 1 family inet address 6.6.6.1/24
```

22. Set the family inet address for the loopback unit 0.

```
[edit]
user@router# set interfaces lo0 unit 0 family inet address 3.3.3.3/32
```

23. Set up OSPF for the ge-8/3/8.0 interface.

```
[edit]
user@router# set protocols ospf area 0.0.0.0 interface ge-8/3/8.0
```

24. Configure OSPF for the loopback interface.

```
[edit]
user@router# set protocols ospf area 0.0.0.0 interface lo0.0
```

25. Set up OSPF for the xe-0/1/3.0 interface.

```
[edit]
user@router# set protocols ospf area 0.0.0.0 interface xe-0/1/3.0
```

26. Configure OSPF for the ge-8/3/2.0 interface.

```
[edit]
user@router# set protocols ospf area 0.0.0.0 interface ge-8/3/2.0
```

27. Set up the static address for the physical interface module (PIM) rendezvous point (RP).

```
[edit]
user@router# set protocols pim rp static address 10.2.1.3
```

28. Configure the loopback interface to bidirectional sparse mode for the PIM protocol.

```
[edit]
user@router# set protocols pim interface lo0.0 mode bidirectional-sparse
```


29. Set the ge-8/3/8.0 interface to bidirectional sparse mode for the PIM protocol.

    ```
    [edit]
    user@router# set protocols pim interface ge-8/3/8.0 mode bidirectional-sparse
    ```

30. Configure the xe-0/1/3.0 interface to bidirectional sparse mode for the PIM protocol.

    ```
    [edit]
    user@router# set protocols pim interface xe-0/1/3.0 mode bidirectional-sparse
    ```

31. Set the ge-8/3/2.0 interface to bidirectional sparse mode for the PIM protocol.

    ```
    [edit]
    user@router# set protocols pim interface ge-8/3/2.0 mode bidirectional-sparse
    ```

32. Configure redundant graceful switchover on the chassis.

    ```
    [edit]
    user@router# set chassis redundancy graceful-switchover
    ```

33. Set the aggregated ethernet device count to 10.

    ```
    [edit]
    user@router# set chassis aggregated-devices ethernet device-count 10
    ```

34. Configure the tunnel services bandwidth for FPC 1/PIC 0.

    ```
    [edit]
    user@router# set chassis fpc 1 pic 0 tunnel-services bandwidth 10g
    ```

35. Enable enhanced IP for network services on the chassis.

    ```
    [edit]
    user@router# set chassis network-services enhanced-ip
    ```

Results

From configuration mode, confirm your configuration by entering the following commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

user@router# show switch-options

```
switch-options {
  vtep-source-interface lo0.0;
}
```

user@router# show bridge-domains

```
bridge-domains {
  vlan-5 {
    vxlan {
      vni 100;
      multicast-group 239.1.1.1;
    }
    vlan-id 100;
    routing-interface irb.0;
    interface xe-1/0/0.0;
  }
```

```
    vlan-6 {
      vxlan {
        vni 200;
        multicast-group 239.2.1.1;
      }
      vlan-id 200;
      routing-interface irb.1;
      interface xe-2/0/0.0;
    }
  }
```

user@router# show interfaces

```
  interfaces {
    xe-1/0/0 {
      vlan-tagging;
      encapsulation flexible-ethernet-services;
      unit 0 {
        encapsulation vlan-bridge;
        vlan-id 100;
      }
    }
    xe-2/0/0 {
      vlan-tagging;
      encapsulation flexible-ethernet-services;
      unit 0 {
        encapsulation vlan-bridge;
        vlan-id 200;
      }
    }
    irb {
      unit 0 {
        family inet {
          address 5.5.5.1/24;
        }
      }
      unit 1 {
        family inet {
          address 6.6.6.1/24;
        }
      }
    }
    lo0 {
      unit 0 {
        family inet {
          address 3.3.3.3/32;
        }
      }
    }
  }
```

user@router# show protocols ospf

```
  area 0.0.0.0 {
    interface ge-8/3/8.0;
    interface lo0.0;
    interface xe-0/1/3.0;
    interface ge-8/3/2.0;
```



```
}
```

user@router# show protocols pim

```
rp {
    static {
        address 10.2.1.3;
    }
}
```

user@router# show chassis

```
redundancy {
    graceful-switchover;
}
aggregated-devices {
    ethernet {
        device-count 10;
    }
}
fpc 1 {
    pic 0 {
        tunnel-services {
            bandwidth 10g;
        }
    }
}
network-services enhanced-ip;
```

Verification

Confirm that the configuration is working properly.

- Verifying Reachability on page 55
- Verifying VXLAN on page 56

Verifying Reachability

Purpose Verify that the network is up and running with the proper interfaces and routes installed.

Action

user@router> show interfaces terse irb

```
Interface               Admin Link Proto    Local                 Remote
irb                     up    up
irb.0                   up    up   inet     5.5.5.1/24
                                   multiservice
irb.1                   up    up   inet     6.6.6.1/24
                                   multiservice
```

user@router> ping 5.5.5.1/24

```
PING 5.5.5.1 (5.5.5.1): 56 data bytes
64 bytes from 5.5.5.1: icmp_seq=0 ttl=64 time=0.965 ms
64 bytes from 5.5.5.1: icmp_seq=1 ttl=64 time=0.960 ms
64 bytes from 5.5.5.1: icmp_seq=2 ttl=64 time=0.940 ms
^C
--- 1.1.1.1 ping statistics ---
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max/stddev = 0.940/0.955/0.965/0.011 ms
```

Meaning

Use the **show interfaces terse irb** command to verify that the IRB interface has been properly configured. The **irb.0** and **irb.1** interfaces should display the proper multiservice inet addresses.

Use the **ping** command to confirm that the network is connected to the IRB multiservice address.

Verifying VXLAN

Purpose

Verify that VXLAN is working and the proper protocols are enabled.

Action

user@router> show interfaces vtep

```
Physical interface: vtep, Enabled, Physical link is Up
  Interface index: 133, SNMP ifIndex: 575
  Type: Software-Pseudo, Link-level type: VxLAN-Tunnel-Endpoint, MTU: 1600, Speed:
 Unlimited
  Device flags   : Present Running
  Interface flags: SNMP-Traps
  Link type      : Full-Duplex
  Link flags     : None
  Last flapped   : Never
    Input packets : 0
    Output packets: 0

  Logical interface vtep.32768 (Index 334) (SNMP ifIndex 607)
    Flags: Up SNMP-Traps Encapsulation: ENET2
    VXLAN Endpoint Type: Source, VXLAN Endpoint Address: 10.255.187.32, L2 Routing
 Instance: default-switch, L3 Routing Instance: default
    Input packets : 0
    Output packets: 0
```

user@router> show l2-learning vxlan-tunnel-end-point remote mac-table

```
MAC flags (S -static MAC, D -dynamic MAC, L -locally learned, C -Control MAC
           SE -Statistics enabled, NM -Non configured MAC, R -Remote PE MAC)

Logical system   : <default>
Routing instance : default-switch
 Bridging domain : vlan-5+100, VLAN : 100, VNID : 100
 Bridging domain : vlan-6+200, VLAN : 200, VNID : 200
```

user@router> show l2-learning vxlan-tunnel-end-point source

```
Logical System Name       Id  SVTEP-IP         IFL   L3-Idx
<default>                 0   10.255.187.32    lo0.0    0
    L2-RTT                   Bridge Domain              VNID     MC-Group-IP
    default-switch           vlan-5+100                 100      239.1.1.1

    default-switch           vlan-6+200                 200      239.1.1.1
```

Meaning

Use the **show interface vtep** command to displays information about VXLAN endpoint configuration. Make sure the routing instance is assigned to the default-switch..

Use the **show l2-learning vxlan-tunnel-end-point remote mac-table** command to confirm that the bridging domain VLAN groups were configured correctly.

Use the **show l2-learning vxlan-tunnel-end-point source** command to confirm the multicast IP addresses for bridging domain VLAN groups.

Related Documentation

- Understanding VXLANs on page 41
- *show bridge mac-table*
- *show vpls mac-table*

CHAPTER 6

High Availability for Service VLANs

Ethernet OAM Support for Service VLANs Overview

You can enable propagation of the Ethernet IEEE 802.1ag Operation, Administration, and Maintenance (OAM) state of a static single-tagged service VLAN (S-VLAN) to a dynamic or static double-tagged customer VLAN (C-VLAN) and, by extension, to the subscriber interfaces configured on the C-VLAN. The static S-VLAN logical interface must be configured on a Gigabit Ethernet, 10-Gigabit Ethernet, or aggregated Ethernet physical interface.

Propagation of the S-VLAN OAM state to associated C-VLANs ensures that when the OAM state of the S-VLAN link is down, the associated C-VLANs and all subscriber interfaces configured on the C-VLANs are brought down as well.

Ethernet OAM Support for Service VLANs Terms and Acronyms

Table 3 on page 59 defines the basic terms and acronyms used in this discussion of Ethernet OAM support for service VLANs.

Table 3: Ethernet OAM Support for Service VLANs Terms and Acronyms

Term	Definition
CFM	Connectivity fault management. Provides end-to-end monitoring of an Ethernet network that can be made up of one or more service instances. Junos OS supports Ethernet IEEE 802.1ag CFM.
Continuity check protocol	A feature of Ethernet IEEE 802.1ag CFM that provides fault detection within a maintenance association.

Table 3: Ethernet OAM Support for Service VLANs Terms and Acronyms *(continued)*

Term	Definition
C-VLAN	Customer VLAN. A dynamic or static double-tagged logical interface that has both an outer VLAN tag (corresponding to the S-VLAN) and an inner VLAN tag (corresponding to the C-VLAN). In a 1:1 subscriber network access model, dedicated C-VLANs provide a one-to-one correspondence between an individual subscriber and the VLAN encapsulation.
OAM	Operation, Administration, and Maintenance. A set of Ethernet connectivity specifications and functions providing connectivity monitoring, fault detection and notification, fault verification, fault isolation, loopback, and remote defect identification. Ethernet interfaces on MX Series routers support the IEEE 802.1ag standard for OAM.
S-VLAN	Service VLAN. A static single-tagged logical interface that has only one outer VLAN tag (corresponding to the S-VLAN). In an N:1 subscriber network access model, S-VLANs are dedicated to a particular service, such as video, voice, or data, instead of to a particular subscriber. Because an S-VLAN is typically shared by many subscribers within the same household or in different households, it provides a many-to-one correspondence between individual subscribers and the VLAN encapsulation.
VLAN	Virtual local area network. A logical group of network devices that appear to be on the same local area network, regardless of their physical location.

Components of Ethernet OAM Support for Service VLANs

Ethernet OAM support for S-VLANs involves the following components:

- Physical interface—On MX Series routers with Modular Port Concentrator/Modular Interface Card (MPC/MIC) interfaces, you can enable propagation of the S-VLAN OAM state to a C-VLAN on Gigabit Ethernet, 10-Gigabit Ethernet, or aggregated Ethernet physical interfaces.
- S-VLAN—To enable propagation of the S-VLAN Ethernet OAM state to associated C-VLANs and subscriber interfaces, you must configure the static single-tagged S-VLAN logical interface to run the Ethernet IEEE 802.1ag CFM continuity check protocol.
- C-VLAN—The C-VLAN is a dynamic or static double-tagged logical interface that has the same S-VLAN (outer) tag as the static single-tagged S-VLAN logical interface. If propagation of the S-VLAN OAM state to the C-VLAN is enabled on the physical interface, the router brings down the C-VLAN and its associated subscriber interfaces when the CFM continuity check protocol detects that the OAM state of the underlying S-VLAN is down.
- Subscriber interfaces—Propagation of the S-VLAN Ethernet OAM state to associated C-VLANs and subscriber interfaces applies to all dynamic or static DHCP, IP demultiplexing (IP demux), and PPPoE subscriber interfaces configured on the C-VLAN.

How Ethernet OAM Support for Service VLANs Works

To enable propagation of the Ethernet OAM state of the S-VLAN to associated C-VLANs and subscriber interfaces, use the **oam-on-svlan** statement when you configure a Gigabit Ethernet (ge), 10-Gigabit Ethernet (xe), or aggregated Ethernet (ae) physical interface.

If Ethernet IEEE 802.1ag CFM is properly configured on the S-VLAN logical interface, including the **oam-on-svlan** statement for these Ethernet interfaces causes the router to bring down both of the following when the CFM continuity check protocol detects that the OAM state of the S-VLAN logical interface is down:

- All dynamic or static double-tagged C-VLAN logical interfaces that have the same S-VLAN (outer) tag as the S-VLAN logical interface on which they are configured.
- All dynamic or static DHCP, IP demux, and PPPoE logical subscriber interfaces configured on the associated C-VLANs.

To illustrate how Ethernet OAM support for S-VLANs works, consider the following sample configuration on a Gigabit Ethernet physical interface:

- Gigabit Ethernet physical interface ge-1/0/3 configured with the **svlan-on-oam** statement.
- Static single-tagged S-VLAN logical interface ge-1/0/3.0, which has a single S-VLAN outer tag, VLAN ID 600.
- Ethernet OAM CFM protocol configured on the static S-VLAN logical interface. The CFM configuration includes an action profile with the **interface-down** default action to bring down the C-VLAN and dynamic subscriber interfaces when the continuity check protocol detects that the Ethernet OAM state of S-VLAN interface ge-1/0/3.0 is down.
- Static double-tagged C-VLAN logical interface ge-1/0/3.100, which has an S-VLAN outer tag, VLAN ID 600, and a C-VLAN inner tag, VLAN ID 1.
- Static PPPoE subscriber interfaces configured on C-VLAN interface ge-1/0/3.100.

Because the S-VLAN and C-VLAN logical interfaces in this example have the same S-VLAN outer tag (VLAN ID 600), the router brings down the C-VLAN interface and the PPPoE logical subscriber interfaces when the CFM continuity check detects that the OAM status of S-VLAN interface ge-1/0/3.0 is down.

Restrictions for Using Ethernet OAM Support for Service VLANs

Ethernet OAM support for S-VLANs is *not currently supported* for use with any of the following:

- Dynamically configured S-VLAN logical interfaces
- S-VLAN trunk interfaces
- C-VLAN trunk interfaces

Related Documentation

- Configuring Ethernet OAM Support for Service VLANs with Double-Tagged Customer VLANs on page 62

- *IEEE 802.1ag OAM Connectivity Fault Management Overview*

Configuring Ethernet OAM Support for Service VLANs with Double-Tagged Customer VLANs

On MX Series routers with MPC/MIC interfaces, you can enable propagation of the Ethernet IEEE 802.1ag Operation, Administration, and Maintenance (OAM) state of a static single-tagged service VLAN (S-VLAN) to the dynamic or static double-tagged customer VLAN (C-VLAN) that has the same S-VLAN (outer) tag as the S-VLAN, and, by extension, to subscriber interfaces configured on the C-VLAN. The static S-VLAN logical interface must be configured on a Gigabit Ethernet, 10-Gigabit Ethernet, or aggregated Ethernet physical interface.

Before you begin:

- Make sure the static single-tagged S-VLAN logical interface is configured with the Ethernet 802.1ag OAM connectivity fault management (CFM) continuity check protocol.

  See *IEEE 802.1ag OAM Connectivity Fault Management Overview*.

To enable propagation of the Ethernet OAM state of a static single-tagged S-VLAN to dynamic or static double-tagged C-VLAN logical interfaces:

- Configure a Gigabit Ethernet (ge), 10-Gigabit Ethernet (xe), or aggregated Ethernet (ae) physical interface to propagate the S-VLAN Ethernet OAM state to C-VLAN logical interfaces that have the same S-VLAN (outer) tag as the S-VLAN interface.

  ```
  [edit]
  user@host# set interfaces interface-name-fpc/pic/port oam-on-svlan
  ```

  For example, the following statement enables propagation of the Ethernet OAM state of a static single-tagged S-VLAN on Gigabit Ethernet interface ge-1/0/5 to a dynamic or static double-tagged C-VLAN logical interface with the same S-VLAN (outer) tag as the S-VLAN interface.

  ```
  [edit]
  user@host# set interfaces ge-1/0/5 oam-on-svlan
  ```

  Including the **oam-on-svlan** statement when you configure a Gigabit Ethernet, 10-Gigabit Ethernet, or aggregated Ethernet physical interface causes the router to bring down both of the following when the CFM continuity check protocol detects that the OAM state of the S-VLAN logical interface is down:

  - All dynamic or static double-tagged C-VLANs on the S-VLAN interface that have the same S-VLAN (outer) tag as the S-VLAN interface.
  - All DHCP, IP demultiplexing (IP demux), and PPPoE logical subscriber interfaces configured on the associated C-VLANs.

Example: Gigabit Ethernet Interface with Static S-VLAN, Dynamic C-VLAN, and

Dynamic PPPoE Subscriber Interfaces

The following example shows a dynamic subscriber access configuration that uses the **oam-on-svlan** statement on a Gigabit Ethernet interface. This example configures Gigabit Ethernet physical interface ge-1/0/5 with a static single-tagged S-VLAN logical interface (ge-1/0/5.1) that runs the Ethernet 802.1ag OAM CFM continuity check protocol. A dynamic profile named double-vlans creates a dynamic double-tagged C-VLAN interface, and a dynamic profile named pppoe-profile creates dynamic PPPoE subscriber interfaces on the C-VLAN interface. The **oam-on-svlan** statement for ge-1/0/5 propagates the Ethernet OAM state of S-VLAN interface ge-1/0/5.1 to the C-VLAN interface and the dynamic PPPoE subscriber interfaces.

For clarity, the configuration is divided into five steps.

1. Configure a dynamic profile named double-vlans that defines a dynamic double-tagged C-VLAN logical interface.

```
[edit]
dynamic-profiles {
  double-vlans {
    interfaces {
      "$junos-interface-ifd-name" {
        unit "$junos-interface-unit" {
          vlan-tags outer "$junos-stacked-vlan-id" inner "$junos-vlan-id";
          encapsulation ppp-over-ether;
          pppoe-underlying-options {
            dynamic-profile pppoe-profile;
          }
        }
      }
    }
  }
}
```

2. Configure a dynamic profile named pppoe-profile that defines dynamic PPPoE subscriber interfaces on the C-VLAN.

```
[edit]
dynamic-profiles {
  pppoe-profile {
    interfaces {
      pp0 {
        unit "$junos-interface-unit" {
          pppoe-options {
            underlying-interface "$junos-underlying-interface";
            server;
          }
          family inet {
            unnumbered-address lo0.0;
          }
        }
      }
    }
  }
}
```

3. Configure Gigabit Ethernet physical interface ge-1/0/5.

```
[edit]
interfaces {
  ge-1/0/5 {
    description "connect to remote router";
    flexible-vlan-tagging;
    oam-on-svlan;
    unit 1 {
      vlan-id 1;
    }
    auto-configure {
      stacked-vlan-ranges {
        dynamic-profile double-vlans {
          accept any;
          ranges {
            any,any;
          }
        }
      }
    }
  }
  lo0 {
    unit 0 {
      family inet {
        address 100.1.1.1/32 {
          primary;
        }
      }
    }
  }
}
```

The preceding example in Step 3 configures a static, single-tagged S-VLAN logical interface (ge-1/0/5.1) with VLAN ID 1, and references the double-vlans dynamic profile to create a dynamic double-tagged C-VLAN logical interface with S-VLAN (outer) tag **any** and C-VLAN (inner) tag **any**. The tag value **any** represents the entire range of VLAN IDs or S-VLAN IDs, including VLAN ID 1.

Because the C-VLAN outer tag (**any**) matches the S-VLAN tag VLAN ID 1, the **oam-on-svlan** statement in the configuration causes the router to propagate the Ethernet OAM state of S-VLAN ge-1/0/5.1 to the dynamic double-tagged C-VLAN logical interface (created by the double-vlans dynamic profile) and, by extension, to the dynamic PPPoE subscriber interfaces on the C-VLAN (created by the pppoe-profile dynamic profile).

4. Configure the Ethernet 802.1ag OAM CFM continuity check protocol on the static S-VLAN interface (ge-1/0/5.1).

```
[edit]
protocols {
  oam {
    ethernet {
      connectivity-fault-management {
        action-profile myDefault {
          default-actions {
```



```
              interface-down;
            }
          }
          maintenance-domain md1 {
            level 1;
            maintenance-association ma1 {
              continuity-check {
                interval 1s;
              }
              mep 100 {
                interface ge-1/0/5.1;
                direction down;
                remote-mep 101 {
                  action-profile myDefault;
                }
              }
            }
          }
        }
      }
    }
  }
```

If the CFM continuity check protocol detects that the Ethernet OAM state of S-VLAN interface ge-1/0/5.1 is down, the **interface-down** action in the myDefault action profile causes the router to bring down both of the following:

- The dynamic double-tagged C-VLAN logical interface that has the same S-VLAN (outer) tag as S-VLAN interface ge-1/0/5.1
- The dynamic PPPoE subscriber interfaces configured on the dynamic C-VLAN interface

5. Create a PPP access profile.

   For brevity, this configuration is only partially shown. The missing portions of the configuration are replaced with ellipses (...).

```
[edit]
access {
  ...
  profile ppp-authenticator {
    ...
  }
}
```

Related Documentation

- Ethernet OAM Support for Service VLANs Overview on page 59
- *IEEE 802.1ag OAM Connectivity Fault Management Overview*

PART 2

Configuring DHCP Subscriber Interfaces

CHAPTER 7

VLAN and Demux Subscriber Interfaces Overview

DHCP Subscriber Interface Overview

You can identify subscribers statically or dynamically.

To identify subscribers statically, you can reference a static VLAN interface in a dynamic profile. To identify subscribers dynamically, you create variables for demux interfaces that are dynamically created by DHCP when subscribers log in.

Statically Identifying Subscribers

Before you can configure static subscriber interfaces in a dynamic profile, you must first configure the logical interfaces on the router to which you expect clients to connect. After you have created the static interfaces, you can modify them by using dynamic profiles to apply configuration parameters.

You can also configure subscribers by creating sets of static IP demux interfaces that are not referenced in a dynamic profile.

When configuring the interfaces stanza within a dynamic profile, you use variables to specify the interface name and the logical unit value. When a DHCP subscriber sends a DHCP request to the interface, the dynamic profile replaces the **interface-name** and **unit** variables with the actual interface name and logical unit number of the interface that received the DHCP request. After this association is made, the router configures the interface with any CoS or protocol (that is, IGMP) configuration within the dynamic profile, or applies any input or output filter configuration that you have associated with that dynamic profile.

```
[edit dynamic-profiles]
interfaces interface-name {
  unit logical-unit-number {
    family family {
      address address;
```

```
                filter {
                    input filter-name;
                    output filter-name;
                }
            unnumbered-address interface-name <preferred-source-address address>;
            vlan-id;
        }
        vlan-tagging;
    }
```

Dynamically Identifying Subscribers

You can configure demux interfaces to represent a subscriber interface in a dynamic profile. When a subscriber logs in using a DHCP access method, the demux interface is dynamically created.

You specify variables for the unit number, the name of the underlying interface, and the IP address in the dynamic profile. These variables are replaced with the values that are supplied by DHCP when the subscriber logs in.

Related Documentation

- Static Subscriber Interfaces and VLAN Overview on page 6
- Subscriber Interfaces and Demultiplexing Overview on page 70

Subscriber Interfaces and Demultiplexing Overview

You can create logical subscriber interfaces using static or dynamic demultiplexing interfaces. In addition, you can use either IP demultiplexing interfaces or VLAN demultiplexing interfaces when creating logical subscriber interfaces.

Demultiplexing (demux) interfaces are logical interfaces that share a common, underlying logical interface (in the case of IP demux) or underlying physical interface (in the case of VLAN demux). You can use these interfaces to identify specific subscribers or to separate individual circuits by IP address (IP demux) or VLAN ID (VLAN demux).

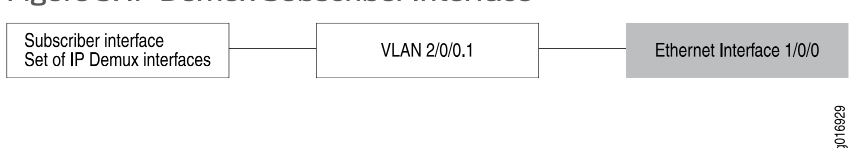
The subscriber interfaces can provide different levels of services for individual subscribers in an access network. For example, you can apply CoS parameters for each subscriber.

Interface Sets of Static Demux Interfaces

You can group static demux interfaces to create individual subscriber interfaces using interface sets. Interface sets enable you to provide the same level of service for a group of subscribers; for example, all residential subscribers who receive the basic data service.

Figure 3 on page 70 shows a subscriber interface configured using a set of IP demux interfaces with an underlying VLAN interface.

Figure 3: IP Demux Subscriber Interface



Dynamic Demultiplexing Interfaces

You can configure demux interfaces to represent a dynamic subscriber interface in a dynamic profile.

Demux interfaces are dynamically created by a DHCP access method when the underlying interface for the demux interface is configured for the access method. The DHCP access model creates the demux interface with the subscriber's assigned IP address (for IP demux interfaces) or VLAN ID (for VLAN demux interfaces).

To configure an IP demux interface in the dynamic profile, you specify variables for the unit number, the name of the underlying interface, and the IP address. To configure a VLAN demux interface in the dynamic profile, you specify variables for the unit number, the name of the underlying interface, and the VLAN ID. These variables are replaced with the values that are supplied by DHCP when the subscriber logs in.

Guidelines for Configuring Demux Interfaces for Subscriber Access

When you configure static or dynamic demux interfaces for subscriber access, consider the following guidelines:

- You can only configure interface sets of static demux interfaces and dynamic demux interfaces on MX Series 3D Universal Edge Routers. Hierarchical and per-unit scheduling is supported for dynamically created demux interfaces on the EQ DPC.
- IP demux interfaces support IPv4 (**family inet**) and IPv6 (**family inet6)**).
- IP demux subscriber interfaces over aggregated Ethernet physical interfaces are supported only for MX Series routers that have only MPCs installed. If the router has other cards in addition to MPCs, the CLI accepts the configuration but errors are reported when the subscriber interfaces are brought up.
- You can configure IPv4 and IPv6 addressing for static and dynamic demux interfaces.
- You can configure only one **demux0** interface per chassis.
- For IP demux interfaces, you can define logical demux interfaces on top of the **demux0** interface (for example, **demux0.1**, **demux0.2**, and so on).
- Demux interfaces currently support only Gigabit Ethernet, Fast Ethernet, 10-Gigabit Ethernet, and aggregated Ethernet underlying interfaces.
- You must associate IP demux interfaces with an underlying logical interface.
- You must associate VLAN demux interfaces with an underlying device (physical interface).
- You cannot use a dynamic demux interface to represent multiple subscribers in a dynamic profile attached to an interface. One dynamic demux interface represents one subscriber. Do not configure the **aggregate-clients** option when attaching a dynamic profile to a demux interface for DHCP.

Related Documentation

- *Configuring Static Subscriber Interfaces Using IP Demux Interfaces*
- *Configuring Static Subscriber Interfaces Using VLAN Demux Interfaces*

- Configuring a Subscriber Interface Using a Set of Static IP Demux Interfaces on page 73
- Configuring a Subscriber Interface Using a Set of Static VLAN Demux Interfaces on page 74
- Configuring Dynamic Subscriber Interfaces Using IP Demux Interfaces in Dynamic Profiles on page 77
- Configuring Dynamic Subscriber Interfaces Using VLAN Demux Interfaces in Dynamic Profiles on page 79
- *Demultiplexing Interface Overview*

IP Demux Interfaces over Static or Dynamic VLAN Demux Interfaces

You can configure a router with IP demux interfaces over VLAN demux interfaces. Just as IP demux interfaces demultiplex their underlying VLAN demux interfaces based on IP address, VLAN demux interfaces demultiplex their underlying aggregate Ethernet or Ethernet interfaces based on VLAN ID.

When configuring IP demux interfaces over VLAN demux interfaces, keep the following in mind:

- Only single and dual VLAN tag options are supported as VLAN selectors.
- Both inet and inet6 families are supported.
- All firewall and CoS features are supported.
- Both static and dynamic VLAN demux interface creation is supported.
- Only MPCs are supported.

Related Documentation

- Subscriber Interfaces and Demultiplexing Overview on page 70
- *Distribution of Demux Subscribers in an Aggregated Ethernet Interface*
- Configuring a Static or Dynamic IP Demux Subscriber Interface over Aggregated Ethernet on page 92
- *Configuring VLAN Dynamic Profiles*
- Example: Dynamic IP Demux Subscriber Interfaces over Dynamic VLAN Demux Interfaces on page 80
- Example: Concurrent Configuration of Dynamic DHCP IP Demux and PPPoE Demux Interfaces over the Same VLAN Demux Interface on page 117
- *Aggregated Ethernet Interfaces Overview*

CHAPTER 8

Configuring Sets of Demux Interfaces to Provide Services to a Group of Subscribers

Configuring a Subscriber Interface Using a Set of Static IP Demux Interfaces

You can create logical subscriber interfaces from IP demux interfaces. IP demultiplexing (demux) interfaces are logical interfaces that share a common, underlying logical interface. IP demux interfaces can be used to identify specific subscribers or to separate individual circuits.

You can group individual subscriber interfaces using interface sets to provide the same level of service for a group of subscribers; for example, all residential subscribers who receive the basic data service. Interface sets can be defined as a list of logical interfaces (unit 0, unit 1, and so on).

To configure a group of static IP demux interfaces:

1. Configure the interface set.

```
interfaces {
  interface-set demux-set {
    interface demux0 {
      unit 0;
      unit 1;
      }
    }
  }
```

2. Define the units of the interface set.

```
demux0 {
  unit 0 {
    demux-options {
      underlying-interface ge-2/0/1.1;
    }
    family inet {
      demux-source {
        1.1.1.0/24;
```

```
            }
            address 1.1.1.1/24;
          }
        }
        unit 1 {
          demux-options {
            underlying-interface ge-2/0/1.1;
          }
          family inet {
            demux-source {
              1.1.2.0/24;
            }
            address 1.1.2.1/24;
          }
        }
      }
```

Related Documentation

- Subscriber Interfaces and Demultiplexing Overview on page 70
- *[edit interfaces] Hierarchy Level*

Configuring a Subscriber Interface Using a Set of Static VLAN Demux Interfaces

You can create logical subscriber interfaces from VLAN demux interfaces. VLAN demultiplexing (demux) interfaces are logical interfaces that share a common, underlying physical interface. VLAN demux interfaces can be used to identify specific subscribers or to separate individual circuits.

You can group individual subscriber interfaces using interface sets to provide the same level of service for a group of subscribers; for example, all residential subscribers who receive the basic data service. Interface sets can be defined as a list of logical interfaces (unit 0, unit 1, and so on).

To configure a group of static VLAN demux interfaces:

1. Configure the interface set.

```
interfaces {
  interface-set demux-set {
    interface demux0 {
      unit 0;
      unit 1;
      }
    }
  }
```

2. Define the units of the interface set.

```
demux0 {
  unit 0 {
    vlan-id 10;
    demux-options {
      underlying-interface ge-2/0/1;
    }
    family inet {
```



```
        address 1.1.1.1/24;
      }
    }
    unit 1 {
      vlan-id 20;
      demux-options {
        underlying-interface ge-2/0/1;
      }
      family inet {
        address 1.1.2.1/24;
      }
    }
  }
```

Related Documentation

- *Configuring CoS on a Set of Static IP Demux Interfaces*
- Subscriber Interfaces and Demultiplexing Overview on page 70
- *[edit interfaces] Hierarchy Level*

CHAPTER 9

Configuring Dynamic Demux Interfaces That are Created by DHCP

Configuring Dynamic Subscriber Interfaces Using IP Demux Interfaces in Dynamic Profiles

You can configure dynamic subscriber interfaces using IP demux interfaces.

To enable the dynamic demux interface to be created by DHCP, you configure the demux options in a dynamic profile. Dynamic profiles enable you to dynamically apply configured values (including CoS, IGMP, or filter configuration) to the dynamic interfaces, making them easier to manage.

Before you begin:

- Configure the dynamic profile.

  See *Configuring a Basic Dynamic Profile*.

To configure dynamic subscriber interfaces:

1. Specify that you want to configure the **demux0** interface in the dynamic profile.

   ```
   user@host#  edit dynamic-profiles business-profile interfaces demux0
   ```

2. Configure the unit for the **demux0** interface.

   a. Configure the variable for the unit number of the **demux0** interface.

      The variable is dynamically replaced with the unit number that DHCP supplies when the subscriber logs in.

      ```
      [edit dynamic-profiles business-profile interfaces demux0]
      user@host# edit unit $junos-interface-unit
      ```


b. Configure the variable for the underlying interface of the demux interfaces and specify the **$junos-underlying-interface** variable.

   The variable is dynamically replaced with the underlying interface that DHCP supplies when the subscriber logs in.

   ```
   [edit dynamic-profiles business-profile interfaces demux0 unit
     "$junos-interface-unit"]
   user@host# set demux-options underlying-interface $junos-underlying-interface
   ```

3. Configure the family for the demux interfaces.

   a. Specify that you want to configure the family.

   For IPv4:

   ```
   [edit dynamic-profiles business-profile interfaces demux0 unit
     "$junos-interface-unit"]
   user@host# edit family inet
   ```

   For IPv6:

   ```
   [edit dynamic-profiles business-profile interfaces demux0 unit
     "$junos-interface-unit"]
   user@host# edit family inet6
   ```

   b. Configure the unnumbered address for the family.

   ```
   [edit dynamic-profiles business-profile interfaces demux0 unit
     "$junos-interface-unit" family inet]
   user@host#  set unnumbered-address lo0.0
   ```

   c. Configure the variable for the IP address of the demux interface.

   The variable is dynamically replaced with the IP address that DHCP supplies when the subscriber logs in. For IPv4, use **$junos-subscriber-ip-address**, For IPv6, use **$junos-subscriber-ipv6-address**. For IPv6 multiple address support, use **$junos-subscriber-ipv6-multi-address**.

   ```
   [edit dynamic-profiles business-profile interfaces demux0 unit
     "$junos-interface-unit" family inet]
   user@host# set demux-source $junos-subscriber-ip-address
   ```

Related Documentation

- Subscriber Interfaces and Demultiplexing Overview on page 70
- Configuring MAC Address Validation for Dynamic Subscriber Interfaces on page 132
- Attaching Dynamic Profiles to DHCP Subscriber Interfaces or DHCP Client Interfaces on page 114
- *Example: Configuring Dynamic Subscriber Interfaces on IP Demux Interfaces*

Configuring Dynamic Subscriber Interfaces Using VLAN Demux Interfaces in Dynamic Profiles

You can configure dynamic subscriber interfaces using VLAN demux interfaces.

To enable the dynamic demux interface to be created by DHCP, you configure the demux options in a dynamic profile. Dynamic profiles enable you to dynamically apply configured values (including CoS, IGMP, or filter configuration) to the dynamic interfaces, making them easier to manage.

Before you begin:

- Configure the dynamic profile.

  See *Configuring a Basic Dynamic Profile*.

To configure dynamic subscriber interfaces:

1. Specify that you want to configure the **demux0** interface in the dynamic profile.

   ```
   user@host#  edit dynamic-profiles business-profile interfaces demux0
   ```

2. Configure the unit for the **demux0** interface.

   a. Configure the variable for the unit number of the **demux0** interface.

      The variable is dynamically replaced with the unit number that DHCP supplies when the subscriber logs in.

      ```
      [edit dynamic-profiles business-profile interfaces demux0]
      user@host# edit unit $junos-interface-unit
      ```

   b. Configure the variable for the underlying interface of the demux interfaces by specifying the **$junos-interface-ifd-name** variable.

      The variable is dynamically replaced with the underlying device name that DHCP supplies when the subscriber logs in.

      ```
      [edit dynamic-profiles business-profile interfaces demux0 unit
        “$junos-interface-unit”]
      user@host# set demux-options underlying-interface $junos-interface-ifd-name
      ```

   c. Configure the variable for the VLAN ID.

      ```
      [edit dynamic-profiles business-profile interfaces demux0 unit
        “$junos-interface-unit”]
      user@host# set vlan-id $junos-vlan-id
      ```

3. Configure the family for the demux interfaces.

   a. Specify that you want to configure the family.

      For IPv4:

      ```
      [edit dynamic-profiles business-profile interfaces demux0 unit
        “$junos-interface-unit”]
      user@host# edit family inet
      ```


For IPv6:

```
[edit dynamic-profiles business-profile interfaces demux0 unit
  “$junos-interface-unit”]
user@host# edit family inet6
```

b. Configure the unnumbered address for the family.

```
[edit dynamic-profiles business-profile interfaces demux0 unit
  “$junos-interface-unit” family inet]
user@host#  set unnumbered-address lo0.0
```

Related Documentation

- Subscriber Interfaces and Demultiplexing Overview on page 70
- Configuring MAC Address Validation for Dynamic Subscriber Interfaces on page 132
- Attaching Dynamic Profiles to DHCP Subscriber Interfaces or DHCP Client Interfaces on page 114
- Example: Dynamic IP Demux Subscriber Interfaces over Dynamic VLAN Demux Interfaces on page 80

Example: Dynamic IP Demux Subscriber Interfaces over Dynamic VLAN Demux Interfaces

This example describes how to configure dynamic IP demux interfaces over dynamic VLAN demux interfaces. You can also configure dynamic IP demux interfaces over static VLAN interfaces. For information on how to configure static VLAN interfaces, see *Configuring a Subscriber Interface with a Static VLAN Interface*.

Requirements

Before you begin, make sure to configure either DHCP Relay or DHCP Local Server. For information about configuring either of these components, see *Extended DHCP Relay Agent Overview* or *Extended DHCP Local Server Overview*.

Overview

You can create a subscriber interface using an IP demux interface stacked on a static or dynamic VLAN demux interface. IP demux interfaces are used to uniquely identify subscribers in an access network based on their IP address

Configuration

- Configuring a Dynamic Profile to Dynamically Create Single-Tagged VLANs on page 83
- Configuring a Dynamic Profile to Dynamically Create IP Demux Interfaces on page 84

Preparing a Subscriber Access Interface

CLI Quick Configuration

To quickly configure the aggregated Ethernet interface over which subscribers access the router:

```
[edit]
set chassis aggregated-devices ethernet device-count 1
set interfaces ge-5/0/9 gigether-options 802.3ad ae0
set interfaces ge-5/1/9 gigether-options 802.3ad ae0
set interfaces ae0 flexible-vlan-tagging
set interfaces ae0 auto-configure vlan-ranges dynamic-profile Auto-VLAN-Demux accept
  inet
set interfaces ae0 auto-configure vlan-ranges dynamic-profile Auto-VLAN-Demux ranges
  ranges 500-1000
set interfaces ae0 aggregated-ether-options lacp active
set interfaces ae0 aggregated-ether-options lacp link-protection
```

Step-by-Step Procedure

You must configure an interface over which clients initially access the router. We recommend that you specify the same VLAN tagging for the interface that you expect from incoming clients. This example uses flexible VLAN tagging to simultaneously support transmission of 802.1Q VLAN single-tag and dual-tag frames on logical interfaces on the same Ethernet port.

If you want it to automatically create dynamic VLANs, the interface must include the VLAN range type (single or stacked) and contain any specific ranges you want the VLANs to use.

To configure an interface for subscriber access:

1. Configure the number of aggregated Ethernet interfaces on the router.

   ```
   [edit]
   user@host# set chassis aggregated-devices ethernet device-count 1
   ```

2. Access the physical interface over which you want subscribers to initially access the router.

   ```
   [edit]
   user@host# edit interfaces ge-5/0/9
   ```

3. Specify the aggregated Ethernet interface to which the physical interface belongs.

   ```
   [edit interfaces ge-5/0/9]
   user@host# set gigether-options 802.3ad ae0
   ```

4. Repeat Step 2 and Step 3 for each interface you want to assign to the aggregated Ethernet bundle.

   ```
   [edit]
   user@host# set interfaces ge-5/1/9 gigether-options 802.3ad ae0
   ```

5. Access the aggregated Ethernet interface.

   ```
   [edit]
   user@host# edit interfaces ae0
   ```

6. Specify the VLAN tagging that you want the aggregated Ethernet interfaces to use.

   [edit interfaces ae0]
   user@host# **set vlan-tagging**

7. Edit the **auto-configure** stanza to automatically configure VLANs.

   [edit interfaces ae0]
   user@host# **edit auto-configure**

8. Edit the **vlan-ranges** stanza for single-tagged VLANs.

   [edit interfaces ae0 auto-configure]
   user@host# **edit vlan-ranges**

9. Specify the dynamic VLAN profile that you want the interface to use for dynamically creating single-tagged VLANs.

   [edit interfaces ae0 auto-configure vlan-ranges]
   user@host# **edit dynamic-profile Auto-VLAN-Demux**

10. Specify what VLAN Ethernet packet type the VLAN profile accepts.

    [edit interfaces ae0 auto-configure vlan-ranges dynamic-profile Auto-VLAN-Demux]
    user@host# **set accept inet**

11. Specify the VLAN ranges that you want the dynamic profile to use. The following example specifies a lower VLAN ID limit of 500 and an upper VLAN ID limit of 1000.

    [edit interfaces ae0 auto-configure vlan-ranges dynamic-profile Auto-VLAN-Demux]
    user@host# **set ranges 500-1000**

12. (Optional) Activate the transmission of LACP packets on the aggregated Ethernet interfaces.

    [edit interfaces ae0]
    user@host# **set aggregated-ether-options lacp active**

13. Specify that the aggregated Ethernet interfaces use link protection.

    [edit interfaces ae0]
    user@host# **set aggregated-ether-options lacp link-protection**

Preparing the Loopback Interface

CLI Quick Configuration

To quickly configure the required loopback interface for this example:

[edit]
set interfaces lo0.0 unit 0 family inet address 100.100.100.1/32

Step-by-Step Procedure

You must configure a loopback interface for use as the unnumbered address and preferred source address for dynamically created interfaces.

To configure the required loopback interface for this example:

1. Configure a loopback interface.

   [edit]
   user@host# **edit interfaces lo0.0**

2. Specify that the loopback interface accept inet packets.


```
[edit interfaces lo0 unit 0]
user@host# edit family inet
```

3. Specify the IP address for the loopback interface.

```
[edit interfaces lo0 unit 0 family inet]
user@host# set address 100.100.100.1/32
```

Configuring a Dynamic Profile to Dynamically Create Single-Tagged VLANs

CLI Quick Configuration

To quickly configure the dynamic profile used to dynamically create single-tagged VLANs in the example:

```
[edit]
set dynamic-profiles Auto-VLAN-Demux interfaces demux0 unit $junos-interface-unit
   demux-source inet
set dynamic-profiles Auto-VLAN-Demux interfaces demux0 unit $junos-interface-unit
   proxy-arp
set dynamic-profiles Auto-VLAN-Demux interfaces demux0 unit $junos-interface-unit
   vlan-id $junos-vlan-id
set dynamic-profiles Auto-VLAN-Demux interfaces demux0 unit $junos-interface-unit
   demux options underlying-interface $junos-interface-ifd-name
set dynamic-profiles Auto-VLAN-Demux interfaces demux0 unit $junos-interface-unit
   family inet unnumbered-address lo0.0 preferred source-address 100.100.100.1
```

Step-by-Step Procedure

For dynamic IP demux interfaces to reside on a dynamic VLAN demux interface, the VLAN interface must first exist.

To configure a dynamic profile that automatically creates VLAN interfaces:

1. Create a dynamic profile for automatically creating single-tagged VLAN interfaces.

```
[edit]
user@host# edit dynamic-profiles Auto-VLAN-Demux
```

2. Specify that the dynamic VLAN profile use the demux interface.

```
[edit dynamic-profiles “Auto-VLAN-Demux”]
user@host# edit interfaces demux0
```

3. Specify that the dynamic profile apply the demux interface unit value to the dynamic VLANs.

```
[edit dynamic-profiles Auto-VLAN-Demux interfaces demux0]
user@host# edit unit $junos-interface-unit
```

4. (Optional) Specify that the demux source accepts only IPv4 (inet) packets.

```
[edit dynamic-profiles Auto-VLAN-Demux interfaces demux0 unit
   “$junos-interface-unit”]
user@host# set demux-source inet
```

5. (Optional) Specify that each dynamically created interface respond to any ARP request, as long as an active route exists to the target address of the ARP request.

```
[edit dynamic-profiles Auto-VLAN-Demux interfaces demux0 unit
   “$junos-interface-unit”]
user@host# set proxy-arp
```

6. Specify that VLAN IDs are dynamically created.

   [edit dynamic-profiles Auto-VLAN-Demux interfaces demux0 unit "$junos-interface-unit"]
   user@host# **set vlan-id $junos-vlan-id**

7. Specify the logical underlying interface for the dynamic VLANs.

   [edit dynamic-profiles Auto-VLAN-Demux interfaces demux0 unit "$junos-interface-unit"]
   user@host# **set demux-options underlying-interface $junos-interface-ifd-name**

8. Specify that the VLAN demux interface can accept inet family packets for IPoE/DHCP subscribers.

   [edit dynamic-profiles Auto-VLAN-Demux interfaces demux0 unit "$junos-interface-unit"]
   user@host# **edit family inet**

9. Specify the loopback address as the unnumbered address and preferred source address for the inet family.

   [edit dynamic-profiles Auto-VLAN-Demux interfaces demux0 unit "$junos-interface-unit" family inet]
   user@host# **set unnumbered-address lo0.0 preferred-source-address 100.100.100.1**

Configuring a Dynamic Profile to Dynamically Create IP Demux Interfaces

CLI Quick Configuration

To quickly configure the dynamic profile used to dynamically create IP demux interfaces in the example:

[edit]
set dynamic-profiles DHCP-IP-Demux interfaces demux0 unit $junos-interface-unit proxy-arp
set dynamic-profiles DHCP-IP-Demux interfaces demux0 unit $junos-interface-unit demux-options underlying-interface $junos-underlying-interface
set dynamic-profiles DHCP-IP-Demux interfaces demux0 unit $junos-interface-unit family inet demux-source $junos-subscriber-ip-address
set dynamic-profiles DHCP-IP-Demux interfaces demux0 unit $junos-interface-unit family inet unnumbered-address lo0.0 preferred-source-address 100.100.100.1

Step-by-Step Procedure

To configure a dynamic profile that automatically creates IP demux interfaces:

1. Create a dynamic profile for dynamically creating IP demux interfaces.

   [edit]
   user@host# **edit dynamic-profiles DHCP-IP-Demux**

2. Specify that the dynamic profile use the demux0 interface.

   [edit dynamic-profiles DHCP-IP-Demux]
   user@host# **edit interfaces demux0**

3. Specify that the dynamic profile apply the interface unit value to the dynamic IP demux interfaces.

   [edit dynamic-profiles DHCP-IP-Demux interfaces demux0]
   user@host# **edit unit $junos-interface-unit**

4. (Optional) Configure the router to respond to any ARP request, as long as the router has an active route to the target address of the ARP request.

   [edit dynamic-profiles DHCP-IP-Demux interfaces demux0 unit "$junos-interface-unit"]
   user@host# **set proxy-arp**

5. Specify the logical underlying interface for the dynamic IP demux interfaces.

   [edit dynamic-profiles DHCP-IP-Demux interfaces demux0 unit "$junos-interface-unit"]
   user@host# **set demux-options underlying-interface $junos-underlying-interface**

6. Specify the protocol family information for the dynamic IP demux interfaces.

   [edit dynamic-profiles DHCP-IP-Demux interfaces demux0 unit "$junos-interface-unit"]
   user@host# **edit family inet**

7. Specify the demux source address is obtained from the incoming subscriber IP address.

   [edit dynamic-profiles DHCP-IP-Demux interfaces demux0 unit "$junos-interface-unit" family inet]
   user@host# **set demux-source $junos-subscriber-ip-address**

8. Specify the loopback interface as the unnumbered address and the demux interface IP address as the preferred source address for the dynamic IP demux interfaces.

   [edit dynamic-profiles DHCP-IP-Demux interfaces demux0 unit "$junos-interface-unit" family inet]
   user@host# **set unnumbered-address lo0.0 preferred-source-address 100.100.100.1**

Verification

- Subscriber Verification on page 85
- Interface Verification on page 85

Subscriber Verification

Purpose View subscriber information on the router.

Action

- To display dynamic subscriber information:

  user@host# **show subscribers detail**

Interface Verification

Purpose View interface-specific information on the router.

Action

- To display interface-specific output:

  user@host# **show interfaces *interface-name***

Related Documentation

- *Dynamic Profiles Overview*

- *Configuring a Basic Dynamic Profile*
- *Configuring Predefined Dynamic Variables in Dynamic Profiles*
- Dynamic 802.1Q VLAN Overview on page 5
- *Configuring VLAN Dynamic Profiles*
- *Demultiplexing Interface Overview*

CHAPTER 10

Configuring DHCP Subscriber Interfaces over Aggregated Ethernet

Static and Dynamic VLAN Subscriber Interfaces over Aggregated Ethernet Overview

You can configure a subscriber interface represented by a static virtual LAN (VLAN) stacked on a two-link aggregated Ethernet logical interface. You must configure the aggregated Ethernet logical interface on Enhanced Queuing Dense Port Concentrators (EQ DPCs) or MPC/MIC interfaces in MX Series 3D Universal Edge Routers.

A static or dynamic VLAN subscriber interface over aggregated Ethernet can also support one-to-one active/backup link redundancy, depending on how you configure the underlying aggregated Ethernet interface.

To configure a static or dynamic VLAN subscriber interface over aggregated Ethernet, make sure you understand the following concepts.

- Guidelines for Configuring an Aggregated Ethernet Logical Interface to Support a Static or Dynamic VLAN Subscriber Interface on page 88

Guidelines for Configuring an Aggregated Ethernet Logical Interface to Support a Static or Dynamic VLAN Subscriber Interface

The following guidelines for configuring an aggregated Ethernet logical interface also apply to configuring a static or dynamic VLAN subscriber interface stacked on a two-link aggregated Ethernet logical interface:

- If you need to support one-to-one active/backup link redundancy, configure the aggregated Ethernet interface in link protection mode, which requires that the two underlying physical interfaces be designated as primary and backup links.
- In addition, if you need to support one-to-one active/backup link redundancy at the DPC or MPC level, configure the aggregated Ethernet interface on physical interfaces that reside on different EQ DPCs or MPCs.

NOTE: **One-to-one active/backup DPC redundancy is also supported with firewall filters and policy filters for static non-VLAN interfaces configured on an aggregated Ethernet logical interfaces, provided LACP is not active.**

Related Documentation

- Static Subscriber Interfaces and VLAN Overview on page 6
- Configuring a Static or Dynamic VLAN Subscriber Interface over Aggregated Ethernet on page 91
- Example: Configuring a Static Subscriber Interface on a VLAN Interface over Aggregated Ethernet on page 95

- *Guidelines for Configuring Dynamic CoS for Subscriber Access*
- *CoS for Subscriber Access Overview*

Static or Dynamic Demux Subscriber Interfaces over Aggregated Ethernet Overview

You can configure a subscriber interface using a static or dynamic demux interface stacked on an aggregated Ethernet logical interface. Subscriber interfaces on static or dynamic demux interfaces can be used to identify specific subscribers (authenticated users) in an access network or to separate individual circuits. A subscriber interface on a static or dynamic demux interface over aggregated Ethernet can support one-to-one active/backup link redundancy or traffic load balancing, depending on how you configure the underlying aggregated Ethernet interface.

To configure a static or dynamic demux subscriber interface over aggregated Ethernet, make sure you understand the following concepts:

Options for Aggregated Ethernet Logical Interfaces That Support Demux Subscriber Interfaces

Traffic forwarding through a demux logical interface is dependent on the configuration of the underlying interface. Using an aggregated Ethernet interface as the underlying interface for a static or dynamic demux subscriber interface provides you with the following options:

- **1:1 Active/Backup Link Redundancy**—If you need to support one-to-one active/backup link redundancy, configure the aggregated Ethernet interface in link protection mode, which requires that two underlying physical interfaces be designated as primary and backup links. In addition, if you need to support one-to-one active/backup link redundancy at the line card level, configure the aggregated Ethernet interface on physical interfaces that reside either on different EQ DPCs or on different MPCs. When using LACP link protection, you can configure only two member links to an aggregated Ethernet interface: one active and one standby.
- **Load Balancing**—You can configure load balancing instead of 1:1 active/backup link redundancy. The Junos OS implementation of the IEEE 802.3ad standard balances traffic across the member links within an aggregated Ethernet bundle based on the Layer 3 information carried in the packet.

  By default, the system supports hash-based distribution in load balancing scenarios. In this model, traffic for a logical interface can be distributed over multiple links in the aggregated Ethernet interface. If distribution flows are not even, egress CoS scheduling can be inaccurate. In addition, scheduler resources are required on every link of the aggregated Ethernet interface.

  Targeted distribution enables you to target the egress traffic for IP and VLAN demux subscribers on a single member link, using a single scheduler resource. The system distributes the subscriber interfaces equally among the member links.

Hardware Requirements with Static or Dynamic Demux Subscriber Interfaces over Aggregated Ethernet

IP demux subscriber interfaces over aggregated Ethernet interfaces are supported on EQ DPCs on MX Series routers.

VLAN demux subscriber interfaces over aggregated Ethernet interfaces are supported on MX Series routers that only have MPCs installed. If the router has other line cards in addition to MPCs, the CLI accepts the configuration but errors are reported when the subscriber interfaces are brought up.

Features Supported with Static or Dynamic Demux Subscriber Interfaces over Aggregated Ethernet

Table 4 on page 90 lists key subscriber access features supported with static or dynamic demux subscriber interfaces, organized by type of underlying interface:

- Aggregated Ethernet
- Non-aggregated Ethernet (Gigabit Ethernet, Fast Ethernet, or 10-Gigabit Ethernet)

There are no feature limitations specific to demultiplexing. Instead, demux interfaces over aggregated Ethernet are subject to the same scaling and configuration limitations inherent to aggregated Ethernet logical interfaces.

Table 4: Features Supported with Static or Dynamic Demux Subscriber Interfaces

	Static or Dynamic Demux Subscriber Interface	
Feature	Aggregated Ethernet Underlying Interface	Non-aggregated Underlying Logical Interface
Protocol family support	IPv4, IPv6, and PPPoE	IPv4, IPv6, and PPPoE
Per-subscriber firewall filtering and statistics	Supported	Supported
Hierarchical CoS	Supported	Supported
Per-subscriber CoS parameters within the **[edit dynamic-profiles *profile-name* class-of-service]** hierarchy	Supported	Supported
Per-subscriber IGMP configuration within the **[edit dynamic-profiles *profile-name* protocols]** hierarchy NOTE: IP demux interfaces must use OIF mapping. See *Example: Configuring Multicast with Subscriber VLANs* for additional information.	Yes	Yes

Related Documentation

- Subscriber Interfaces and Demultiplexing Overview on page 70
- *Distribution of Demux Subscribers in an Aggregated Ethernet Interface*

- Configuring a Static or Dynamic IP Demux Subscriber Interface over Aggregated Ethernet on page 92
- Configuring the PPPoE Family for an Underlying Interface on page 159
- Example: Configuring a Static Subscriber Interface on an IP Demux Interface over Aggregated Ethernet on page 98
- *Aggregated Ethernet Interfaces Overview*

Configuring a Static or Dynamic VLAN Subscriber Interface over Aggregated Ethernet

You can configure a subscriber link represented by a static virtual LAN (VLAN) stacked on an aggregated Ethernet logical interface.

You can configure subscriber management services such as firewall filters and CoS for this subscriber interface.

To configure a subscriber interface using a static VLAN interface over an aggregated Ethernet logical interface:

1. Configure the aggregated Ethernet interface.
   a. Configure the number of aggregated Ethernet interfaces on the router.

      See *Configuring the Number of Aggregated Ethernet Interfaces on the Device*.
   b. Configure the aggregated Ethernet interface.

      See *Configuring an Aggregated Ethernet Interface*.
   c. (Optional) Configure LACP.

      See *Configuring LACP for Aggregated Ethernet Interfaces*.
   d. (Optional) Configure the minimum number of links.

      See *Configuring Aggregated Ethernet Minimum Links*.
   e. (Optional) Configure the link speed.

      See *Configuring Aggregated Ethernet Link Speed*.
   f. (Optional) Configure the aggregated Ethernet logical interface to support one-to-one active/backup link redundancy or traffic load balancing.

      See *Configuring Aggregated Ethernet Link Protection*.

      NOTE: Link protection is required if you want to configure hierarchical CoS on the aggregated Ethernet interface. For more information, see *Configuring Hierarchical CoS for a Subscriber Interface of Aggregated Ethernet Links*.
2. Configure the static or dynamic VLAN interface.

- For static VLAN interfaces, see *Configuring a Subscriber Interface with a Static VLAN Interface*.
- For dynamic VLAN interfaces, see *Configuring VLAN Dynamic Profiles* and *Configuring VLAN Interfaces to Use Dynamic Profiles*.

3. Configure subscriber management services on the subscriber interface.
   - For firewall filters, see *Dynamically Attaching Statically Created Filters for Any Interface Type* or *Dynamically Attaching Statically Created Filters for a Specific Interface Family Type*.
   - For hierarchical CoS, see *Configuring Hierarchical CoS for a Subscriber Interface of Aggregated Ethernet Links*.

Related Documentation

- Static and Dynamic VLAN Subscriber Interfaces over Aggregated Ethernet Overview on page 88
- Example: Configuring a Static Subscriber Interface on a VLAN Interface over Aggregated Ethernet on page 95
- *Guidelines for Configuring Dynamic CoS for Subscriber Access*
- *CoS for Subscriber Access Overview*

Configuring a Static or Dynamic IP Demux Subscriber Interface over Aggregated Ethernet

You can configure a subscriber interface using a static or dynamic IP demultiplexing (demux) logical interface stacked on an aggregated Ethernet logical interface. Optionally, you can configure the aggregated Ethernet logical interface to support one-to-one active/backup link redundancy or traffic load balancing.

1. Configure the aggregated Ethernet interface.
   a. Configure the number of aggregated Ethernet interfaces on the router.

      See *Configuring the Number of Aggregated Ethernet Interfaces on the Device*.

   b. Configure the aggregated Ethernet interface.

      See *Configuring an Aggregated Ethernet Interface*.

   c. (Optional) Configure LACP.

      See *Configuring LACP for Aggregated Ethernet Interfaces*.

   d. (Optional) Configure the minimum number of links.

      See *Configuring Aggregated Ethernet Minimum Links*.

   e. (Optional) Configure the link speed.

See *Configuring Aggregated Ethernet Link Speed*.

f. (Optional) Configure the aggregated Ethernet logical interface to support one-to-one active/backup link redundancy or traffic load balancing.

For general instructions, see *Configuring Aggregated Ethernet Link Protection*.

NOTE: Link protection is required if you want to configure hierarchical CoS on the aggregated Ethernet interface. For more information, see *Configuring Hierarchical CoS for a Subscriber Interface of Aggregated Ethernet Links*.

2. Configure the aggregated Ethernet logical interface as the underlying interface to support the static or dynamic IP demux subscriber interface.

   The aggregated Ethernet interface needs to support demultiplexing of incoming traffic to the Ethernet links based on IPv4 destination or source addresses in the incoming packets. In addition, you must configure the IP address of each link.

   See *Configuring an IP Demux Underlying Interface*.

3. Configure the static or dynamic IP demux interface.

   - For static subscriber interfaces, see *Configuring Static Subscriber Interfaces Using IP Demux Interfaces*.
   - For dynamic subscriber interfaces, see "Configuring Dynamic Subscriber Interfaces Using IP Demux Interfaces in Dynamic Profiles" on page 77.

   NOTE: IP demux interfaces currently support only the Internet Protocol version 4 (IPv4) suite (**family inet**).

4. (Optional) Configure subscriber management services on the subscriber interface.

   - For firewall filters, see *Dynamically Attaching Statically Created Filters for Any Interface Type* or *Dynamically Attaching Statically Created Filters for a Specific Interface Family Type*.
   - For hierarchical CoS, see *Configuring Hierarchical CoS for a Subscriber Interface of Aggregated Ethernet Links*.

Related Documentation

- Subscriber Interfaces and Demultiplexing Overview on page 70
- Static or Dynamic Demux Subscriber Interfaces over Aggregated Ethernet Overview on page 89
- Example: Configuring a Static Subscriber Interface on an IP Demux Interface over Aggregated Ethernet on page 98
- *Configuring the Distribution Type for Demux Subscribers on Aggregated Ethernet Interfaces*

Configuring a Static or Dynamic VLAN Demux Subscriber Interface over Aggregated Ethernet

You can configure a subscriber interface using a static or dynamic VLAN demultiplexing (demux) logical interface stacked on an aggregated Ethernet physical interface.

1. Configure the aggregated Ethernet interface.

   a. Configure the number of aggregated Ethernet interfaces on the router.

      See *Configuring the Number of Aggregated Ethernet Interfaces on the Device*.

   b. Configure the aggregated Ethernet interface.

      See *Configuring an Aggregated Ethernet Interface*.

   c. (Optional) Configure LACP.

      See *Configuring LACP for Aggregated Ethernet Interfaces*.

   d. (Optional) Configure the minimum number of links.

      See *Configuring Aggregated Ethernet Minimum Links*.

   e. (Optional) Configure the link speed.

      See *Configuring Aggregated Ethernet Link Speed*.

   f. (Optional) Configure the aggregated Ethernet logical interface to support one-to-one active/backup link redundancy or traffic load balancing.

      For general instructions, see *Configuring Aggregated Ethernet Link Protection*.

2. Configure the aggregated Ethernet physical interface as the underlying interface to support the static or dynamic VLAN demux subscriber interface.

   The aggregated Ethernet interface needs to support demultiplexing of incoming traffic to the Ethernet links based on the VLAN ID in the incoming packets.

   See *Configuring a VLAN Demux Underlying Interface*.

3. Configure the static or dynamic VLAN demux interface.

   - For static subscriber interfaces, see *Configuring Static Subscriber Interfaces Using VLAN Demux Interfaces*.
   - For dynamic subscriber interfaces, see "Configuring Dynamic Subscriber Interfaces Using VLAN Demux Interfaces in Dynamic Profiles" on page 79.

NOTE: VLAN demux interfaces currently support the Internet Protocol version 4 (IPv4) suite (**family inet**) and the Internet Protocol version 6 (IPv6) suite (**family inet6**).

VLAN demux subscriber interfaces over aggregated Ethernet physical interfaces are supported only for MX Series routers that have only MPCs installed. If the router has other cards in addition to MPCs, the CLI accepts the configuration but errors are reported when the subscriber interfaces are brought up.

4. (Optional) Configure subscriber management services on the subscriber interface.
   - For firewall filters, see *Dynamically Attaching Statically Created Filters for Any Interface Type* or *Dynamically Attaching Statically Created Filters for a Specific Interface Family Type*.
   - For hierarchical CoS, see *Configuring Hierarchical CoS for a Subscriber Interface of Aggregated Ethernet Links*.

Related Documentation

- Subscriber Interfaces and Demultiplexing Overview on page 70
- Static or Dynamic Demux Subscriber Interfaces over Aggregated Ethernet Overview on page 89
- *Associating VLAN IDs to VLAN Demux Interfaces*
- *Example: Configuring IPv4 Static VLAN Demux Interfaces over a Gigabit Ethernet Underlying Interface with DHCP Local Server*
- Example: Configuring IPv4 Static VLAN Demux Interfaces over an Aggregated Ethernet Underlying Interface with DHCP Local Server on page 100
- *Example: Configuring IPv4 Dynamic VLAN Demux Interfaces over a Gigabit Ethernet Underlying Interface with DHCP Local Server*
- Example: Configuring IPv4 Dynamic VLAN Demux Interfaces over an Aggregated Ethernet Underlying Interface with DHCP Local Server on page 103

Example: Configuring a Static Subscriber Interface on a VLAN Interface over Aggregated Ethernet

This example shows how you can configure a subscriber interface using a static virtual LAN (VLAN) stacked on a two-link aggregated Ethernet logical interface. In this example, the underlying aggregated Ethernet logical interface is configured for one-to-one active/backup redundancy at the DPC level, and per-subscriber static hierarchical class-of-service (CoS) is configured by applying CoS parameters at the aggregated Ethernet logical interface.

1. Define the number of aggregated Ethernet interfaces on the router.

In this example, only one aggregated Ethernet logical interface is configured on the router.

```
[edit]
chassis {
  aggregated-devices {
    ethernet {
      device-count 1;
    }
  }
}
```

2. Configure **ae0**, a two-link aggregated Ethernet logical interface to serve as the underlying interface for the static VLAN subscriber interface. In order to support hierarchical CoS, the physical ports must be on EQ DPCs in MX Series routers.

   In this example, the LAG bundle is configured for one-to-one active/backup link redundancy. To support link redundancy at the DPC level, the LAG bundle attaches ports from two different EQ DPCs.

```
[edit]
interfaces {
  ge-5/0/3 {
    gigether-options {
      802.3ad {
        ae0;
        primary;
      }
    }
    ge-5/1/2 {
      gigether-options {
        802.3ad {
          ae0;
          backup;
        }
      }
    }
  }
}
```

3. Configure **ae0** to serve as the underlying interface for the static VLAN interface.

```
[edit]
interfaces {
  ae0 {
    hierarchical-scheduler;
    aggregated-ether-options {
      link-protection;
      minimum-links 1;
      link-speed 1g;
      lacp {
        active;
      }
    }
  }
}
```


4. Configure static traffic-shaping and scheduling parameters.

```
[edit]
class-of-service {
  forwarding-classes { # Associate queue numbers with class names
    queue 0 be;
    queue 1 e;
    queue 2 af;
    queue 3 nc;
  }
  schedulers { # Define output queue properties
    scheduler_be {
      transmit-rate percent 30;
      buffer-size percent 30;
    }
    scheduler_ef {
      transmit-rate percent 40;
      buffer-size percent 40;
    }
    scheduler_af {
      transmit-rate percent 25;
      buffer-size percent 25;
    }
    scheduler_nc {
      transmit-rate percent 5;
      buffer-size percent 5;
    }
  }
  scheduler-maps { # Associate queues with schedulers
    smap_2 {
      forwarding-class be scheduler_be;
      forwarding-class ef scheduler_ef;
      forwarding-class-af scheduler_af;
      forwarding-class-nc scheduler_nc;
    }
  }
}
```

5. Attach static CoS to the physical and logical interfaces of the aggregated Ethernet interface.

   In this example, three traffic control profiles are defined, but only two profiles are applied to the static VLAN subscriber interface over aggregated Ethernet:

   - The **tcp_for_ae_device_pir_500m** profile defines a shaping rate, and it is applied to both of the underlying physical interfaces (**ge-5/0/3** and **ge-5/1/2**).
   - The **tcp-for-ae_smap_video_pir_20m_delay_30m** profile defines a scheduler map, a shaping rate, and a delay buffer rate, and it is applied to one of the logical interfaces on the aggregated Ethernet bundle (**ae0.0**).

```
[edit]
class-of-service {
  traffic-control-profiles { # Configure traffic shaping and scheduling profiles
    tcp_for_ae_device_pir_500m {
      shaping-rate 20m;
```

```
        }
        tcp_for_ae_smap_video_pir_20m_delay_30m {
            scheduler-map smap_video;
            shaping-rate 20m;
            delay-buffer-rate 30m;
        }
        tcp_for_ae_smap_video_cir_50m_delay_75m {
            scheduler-map smap_video;
            guaranteed-rate 50m;
            delay-buffer-rate 75m;
        }
    }
    interfaces { # Apply two traffic-control profiles to the LAG
        ae0 { # Two underlying physical interfaces on separate EQ DPCs
            output-traffic-control-profile tcp-for-ae_device_pir_500m;
            unit 0 { # One of the two logical interfaces on 'ae0'
                output-traffic-control-profile tcp-for-ae_smap_video_pir_20m_delay_30m;
            }
        }
    }
}
```

Related Documentation

- Static and Dynamic VLAN Subscriber Interfaces over Aggregated Ethernet Overview on page 88
- Configuring a Static or Dynamic VLAN Subscriber Interface over Aggregated Ethernet on page 91

- *Guidelines for Configuring Dynamic CoS for Subscriber Access*
- *CoS for Subscriber Access Overview*

Example: Configuring a Static Subscriber Interface on an IP Demux Interface over Aggregated Ethernet

This example shows how you can configure a subscriber interface using a static IP demultiplexing (demux) interface stacked on a two-link aggregated Ethernet logical interface. In this example, the underlying aggregated Ethernet logical interface is configured for one-to-one active/backup redundancy at the DPC level.

1. Define the number of aggregated Ethernet interfaces on the router.

   In this example, only one aggregated Ethernet logical interface is configured on the router:

   ```
   [edit]
   chassis {
       aggregated-devices {
           ethernet {
               device-count 1;
           }
       }
   }
   ```


2. Configure **ae0**, a two-link aggregated Ethernet logical interface to serve as the underlying interface for the static IP demux subscriber interface.

   In this example, the LAG bundle is configured for one-to-one active/backup link redundancy. To support link redundancy at the DPC level, the LAG bundle attaches ports from two different EQ DPCs.

```
[edit]
interfaces {
   ge-5/0/3 {
      gigether-options {
         802.3ad {
            ae0;
            primary;
         }
      }
   }
   ge-5/1/2 {
      gigether-options {
         802.3ad {
            ae0;
            backup;
         }
      }
   }
}
```

3. Configure the aggregated Ethernet logical interface with link protection enabled, and specify the logical demultiplexing source family type for both the active and backup links.

```
[edit]
interfaces {
   ae0 {
      aggregated-ether-options {
         link-protection;
         minimum-links 1;
         link-speed 1g;
      }
      unit 0 {
         demux-source inet {
         family inet {
            address 20.1.1.0/24;
         }
      }
      unit 1 {
         demux-source inet {
         family inet {
            address 20.1.1.1/24;
         }
      }
   }
}
```

4. Configure the IP demux interface over the aggregated Ethernet logical interface.

```
[edit]
```



```
interfaces {
  demux0 {
    unit 101 {
      demux-options {
        underlying-interface ae0.0;
      }
      family inet {
        demux-source 10.1.0.0/16;
        address 1.1.1.0/24;
      }
    }
    unit 101 {
      demux-options {
        underlying-interface ae0.1;
      }
      family inet {
        demux-source 10.1.0.1/16;
        address 1.1.1.1/24;
      }
    }
  }
}
```

Related Documentation

- Subscriber Interfaces and Demultiplexing Overview on page 70
- Static or Dynamic Demux Subscriber Interfaces over Aggregated Ethernet Overview on page 89
- Configuring a Static or Dynamic IP Demux Subscriber Interface over Aggregated Ethernet on page 92

Example: Configuring IPv4 Static VLAN Demux Interfaces over an Aggregated Ethernet Underlying Interface with DHCP Local Server

This example shows how to configure a static IPv4 VLAN demux interface with aggregated Ethernet as the underlying interface. DHCP Local Server configuration enables the association of subscribers to the VLAN demux interface by listing the aggregated Ethernet interface in the DHCP local server configuration.

To configure dynamic subscribers on VLAN demux interfaces:

1. Enable hierarchical scheduling and VLAN tagging on the underlying interface that you plan to use for any VLAN demux interfaces.

```
interfaces {
  ae1 {
    hierarchical-scheduler;
    vlan-tagging;
    aggregated-ether-options {
      minimum-links 1;
      lacp {
        active;
        periodic slow;
        link-protection {
```

```
                non-revertive;
              }
            }
          }
        }
      }
```

2. Define the gigabit Ethernet interfaces that are part of the aggregated Ethernet interface.

```
interfaces {
  ge-5/0/0 {
    gigether-options {
      802.3ad ae1;
    }
  }
  ge-5/2/0 {
    gigether-options {
      802.3ad ae1;
    }
  }
}
```

3. Define the demux interface.

```
interfaces {
  demux0 {
    unit 102 {
      proxy-arp;
      vlan-id 103;
      demux-options {
        underlying-interface ae1;
      }
      family inet {
        unnumbered-address lo0.0 preferred-source-address 173.16.1.1;
      }
    }
  }
}
```

4. Define the loopback interface.

```
interfaces {
  lo0 {
    unit 0 {
      family inet {
        address 192.16.1.1/32;
      }
    }
  }
}
```

5. Configure a dynamic profile for initial subscriber access.

```
dynamic-profiles {
  user-profile {
    interfaces {
      "$junos-interface-ifd-name" {
        unit "$junos-underlying-interface-unit" {
          family inet;
```

```
                }
              }
            }
            protocols {
              igmp {
                interface "$junos-interface-name" {
                  version 3;
                  immediate-leave;
                  promiscuous-mode;
                }
              }
            }
          }
        }
```

6. Configure the access method used to dynamically create the subscriber interfaces.

   The following stanza specifies the aggregated Ethernet interface (**ae1.0**) for use with the dynamically created subscriber interfaces.

```
system {
  services {
    dhcp-local-server {
      group myDhcpGroup {
        authentication {
          password test;
          username-include {
            user-prefix igmp-user1;
          }
        }
        dynamic-profile user-profile;
        interface ae1.0;
      }
    }
  }
}
```

   Instead of using the aggregated Ethernet interface, you can alternatively specify the specific demux interface (**demux0.102**) as the device to use with the subscriber interfaces as follows:

```
system {
  services {
    dhcp-local-server {
      group myDhcpGroup {
        authentication {
          password test;
          username-include {
            user-prefix igmp-user1;
          }
        }
        dynamic-profile user-profile;
        interface demux0.102;
      }
    }
  }
}
```


Related Documentation

- Configuring Dynamic Subscriber Interfaces Using IP Demux Interfaces in Dynamic Profiles on page 77
- Attaching Dynamic Profiles to DHCP Subscriber Interfaces or DHCP Client Interfaces on page 114

Example: Configuring IPv4 Dynamic VLAN Demux Interfaces over an Aggregated Ethernet Underlying Interface with DHCP Local Server

This example shows how to configure the dynamic creation of IPv4 VLAN demux interfaces with aggregated Ethernet as the underlying interface. DHCP Local Server configuration enables the association of subscribers to the VLAN demux interface by listing the aggregated Ethernet interface in the DHCP local server configuration.

NOTE: VLAN demux subscriber interfaces over aggregated Ethernet physical interfaces are supported only for MX Series routers that have only MPCs installed. If the router has other cards in addition to MPCs, the CLI accepts the configuration but errors are reported when the subscriber interfaces are brought up.

To configure dynamic subscribers on dynamic VLAN demux interfaces:

1. Enable VLAN tagging and VLAN auto-configuration on the underlying aggregated Ethernet interface that you plan to use for dynamically created VLAN demux interfaces.

```
interfaces {
  ae1 {
    vlan-tagging;
    auto-configure {
      vlan-ranges {
        dynamic-profile auto-vlanDemux-profile {
          accept inet;
          ranges {
            any;
          }
        }
      }
    }
    aggregated-ether-options {
      minimum-links 1;
      lacp {
        active;
        periodic slow;
        link-protection {
          non-revertive;
        }
      }
    }
  }
}
```

2. Define the gigabit Ethernet interfaces that are part of the aggregated Ethernet interface.

```
interfaces {
  ge-5/0/0 {
    gigether-options {
      802.3ad ae1;
    }
  }
  ge-5/2/0 {
    gigether-options {
      802.3ad ae1;
    }
  }
}
```

3. Define the loopback interface.

```
interfaces {
  lo0 {
    unit 0 {
      family inet {
        address 192.16.1.1/32;
      }
    }
  }
}
```

4. Configure a dynamic profile for subscriber access.

```
dynamic-profiles {
  user-profile {
    interfaces {
      "$junos-interface-ifd-name" {
        unit "$junos-underlying-interface-unit" {
          family inet;
        }
      }
    }
  }
}
```

5. Configure a dynamic profile for VLAN demux interface creation.

```
dynamic-profiles {
  auto-vlanDemux-profile {
    interfaces {
      demux0 {
        unit "$junos-interface-unit" {
          vlan-id "$junos-vlan-id";
          demux-options {
            underlying-interface "$junos-interface-ifd-name";
          }
          family inet {
            filter {
              input rate_limit;
              output rate_limit;
            }
            unnumbered-address lo0.0 preferred-source-address 192.16.1.1;
```



```
                }
              }
            }
          }
        }
      }
```

6. Configure the access method used to dynamically create the subscriber interfaces. The following stanza specifies the aggregated Ethernet interface (**ae1.0**) for use with the dynamically created subscriber interfaces.

```
system {
  services {
    dhcp-local-server {
      group myDhcpGroup {
        authentication {
          password test;
          username-include {
            user-prefix igmp-user1;
          }
        }
        dynamic-profile user-profile;
        interface ae1.0;
      }
    }
  }
}
```

   Instead of using the aggregated Ethernet interface, you can alternatively specify **demux0** as the device to use with the subscriber interfaces as follows:

   NOTE: Because the demux interfaces and unit values are created dynamically, the unit number is not specified for the demux0 interface.

```
system {
  services {
    dhcp-local-server {
      group myDhcpGroup {
        authentication {
          password test;
          username-include {
            user-prefix igmp-user1;
          }
        }
        dynamic-profile user-profile;
        interface demux0;
      }
    }
  }
}
```

Related Documentation

- Configuring Dynamic Subscriber Interfaces Using VLAN Demux Interfaces in Dynamic Profiles on page 79

• Attaching Dynamic Profiles to DHCP Subscriber Interfaces or DHCP Client Interfaces on page 114

Example: Configuring IPv6 Dynamic VLAN Demux Interfaces over an Aggregated Ethernet Underlying Interface with DHCP Local Server

This example shows how to configure the dynamic creation of IPv6 VLAN demux interfaces with aggregated Ethernet as the underlying interface. DHCP Local Server configuration enables the association of subscribers to the VLAN demux interface by listing the aggregated Ethernet interface in the DHCP local server configuration.

NOTE: VLAN demux subscriber interfaces over aggregated Ethernet physical interfaces are supported only for MX Series routers that have only MPCs installed. If the router has other cards in addition to MPCs, the CLI accepts the configuration but errors are reported when the subscriber interfaces are brought up.

To configure dynamic subscribers on dynamic VLAN demux interfaces:

1. Enable VLAN tagging and VLAN auto-configuration on the underlying aggregated Ethernet interface that you plan to use for dynamically created VLAN demux interfaces.

```
interfaces {
  ae1 {
    vlan-tagging;
    auto-configure {
      vlan-ranges {
        dynamic-profile auto-vlanDemux-profile {
          accept inet6;
          ranges {
            any;
          }
        }
      }
    }
    aggregated-ether-options {
      minimum-links 1;
      lacp {
        active;
        periodic slow;
        link-protection {
          non-revertive;
        }
      }
    }
  }
}
```

2. Define the gigabit Ethernet interfaces that are part of the aggregated Ethernet interface.

```
interfaces {
  ge-5/0/0 {
```



```
        gigether-options {
            802.3ad ae1;
        }
    }
    ge-5/2/0 {
        gigether-options {
            802.3ad ae1;
        }
    }
}
```

3. Define the loopback interface.

```
interfaces {
    lo0 {
        unit 0 {
            family inet6 {
                address 2009:174:1:1::1/128;
            }
        }
    }
}
```

4. Configure a dynamic profile for subscriber access.

```
dynamic-profiles {
    user-profile {
        interfaces {
            "$junos-interface-ifd-name" {
                unit "$junos-underlying-interface-unit" {
                    family inet6;
                }
            }
        }
    }
}
```

5. Configure a dynamic profile for VLAN demux interface creation.

```
dynamic-profiles {
    auto-vlanDemux-profile {
        interfaces {
            demux0 {
                unit "$junos-interface-unit" {
                    vlan-id "$junos-vlan-id";
                    demux-options {
                        underlying-interface "$junos-interface-ifd-name";
                    }
                    family inet6 {
                        filter {
                            input v6_rate_limit;
                            output v6_rate_limit;
                        }
                        unnumbered-address lo0.0 preferred-source-address 2009:174:1:1::1;
                    }
                }
            }
        }
```

```
    }
  }
```

6. Configure the access method used to dynamically create the subscriber interfaces. The following stanza specifies the aggregated Ethernet interface (**ae1.0**) for use with the dynamically created subscriber interfaces.

```
system {
  services {
    dhcp-local-server {
      dhcpv6 {
        group myV6DhcpGroup {
          authentication {
            password test;
            username-include {
              user-prefix igmp-user1;
            }
          }
          dynamic-profile user-profile;
          interface ae1.0;
        }
      }
    }
  }
}
```

Instead of using the aggregated Ethernet interface, you can alternatively specify **demux0** as the device to use with the subscriber interfaces as follows:

NOTE: Because the demux interfaces and unit values are created dynamically, the unit number is not specified for the demux0 interface.

```
system {
  services {
    dhcp-local-server {
      dhcpv6 {
        group myV6DhcpGroup {
          authentication {
            password test;
            username-include {
              user-prefix igmp-user1;
            }
          }
          dynamic-profile user-profile;
          interface demux0;
        }
      }
    }
  }
}
```

Related Documentation

- Configuring Dynamic Subscriber Interfaces Using VLAN Demux Interfaces in Dynamic Profiles on page 79

- Attaching Dynamic Profiles to DHCP Subscriber Interfaces or DHCP Client Interfaces on page 114

Example: Configuring IPv4 Dynamic Stacked VLAN Demux Interfaces over an Aggregated Ethernet Underlying Interface with DHCP Local Server

This example shows how to configure the dynamic creation of IPv4 stacked VLAN demux interfaces with aggregated Ethernet as the underlying interface. DHCP Local Server configuration enables the association of subscribers to the VLAN demux interface by listing the aggregated Ethernet interface in the DHCP local server configuration.

NOTE: **VLAN demux subscriber interfaces over aggregated Ethernet physical interfaces are supported only for MX Series routers that have only MPCs installed. If the router has other cards in addition to MPCs, the CLI accepts the configuration but errors are reported when the subscriber interfaces are brought up.**

To configure dynamic subscribers on dynamic VLAN demux interfaces:

1. Enable VLAN tagging and VLAN auto-configuration on the underlying aggregated Ethernet interface that you plan to use for dynamically created VLAN demux interfaces.

```
interfaces {
  ae1 {
    flexible-vlan-tagging;
    auto-configure {
      stacked-vlan-ranges {
        dynamic-profile auto-vlanDemux-profile {
          accept inet;
          ranges {
            any;
          }
        }
      }
    }
    aggregated-ether-options {
      minimum-links 1;
      lacp {
        active;
        periodic slow;
        link-protection {
          non-revertive;
        }
      }
    }
  }
}
```

2. Define the gigabit Ethernet interfaces that are part of the aggregated Ethernet interface.

```
interfaces {
  ge-5/0/0 {
```

```
            gigether-options {
                802.3ad ae1;
            }
        }
        ge-5/2/0 {
            gigether-options {
                802.3ad ae1;
            }
        }
    }
```

3. Define the loopback interface.

```
    interfaces {
        lo0 {
            unit 0 {
                family inet {
                    address 192.16.1.1/32;
                }
            }
        }
    }
```

4. Configure a dynamic profile for subscriber access.

```
    dynamic-profiles {
        user-profile {
            interfaces {
                "$junos-interface-ifd-name" {
                    unit "$junos-underlying-interface-unit" {
                        family inet;
                    }
                }
            }
        }
    }
```

5. Configure a dynamic profile for VLAN demux interface creation.

```
    dynamic-profiles {
        auto-vlanDemux-profile {
            interfaces {
                demux0 {
                    unit "$junos-interface-unit" {
                        vlan-tags outer "$junos-stacked-vlan-id" inner "$junos-vlan-id";
                        demux-options {
                            underlying-interface "$junos-interface-ifd-name";
                        }
                        family inet {
                            filter {
                                input rate_limit;
                                output rate_limit;
                            }
                            unnumbered-address lo0.0 preferred-source-address 192.16.1.1;
                        }
                    }
                }
            }
```



```
    }
  }
```

6. Configure the access method used to dynamically create the subscriber interfaces. The following stanza specifies the aggregated Ethernet interface (**ae1.0**) for use with the dynamically created subscriber interfaces.

```
system {
  services {
    dhcp-local-server {
      group myDhcpGroup {
        authentication {
          password test;
          username-include {
            user-prefix igmp-user1;
          }
        }
        dynamic-profile user-profile;
        interface ae1.0;
      }
    }
  }
}
```

Instead of using the aggregated Ethernet interface, you can alternatively specify **demux0** as the device to use with the subscriber interfaces as follows:

NOTE: Because the demux interfaces and unit values are created dynamically, the unit number is not specified for the demux0 interface.

```
system {
  services {
    dhcp-local-server {
      group myDhcpGroup {
        authentication {
          password test;
          username-include {
            user-prefix igmp-user1;
          }
        }
        dynamic-profile user-profile;
        interface demux0;
      }
    }
  }
}
```

Related Documentation

- Configuring Dynamic Subscriber Interfaces Using VLAN Demux Interfaces in Dynamic Profiles on page 79
- Attaching Dynamic Profiles to DHCP Subscriber Interfaces or DHCP Client Interfaces on page 114

CHAPTER 11

Using Dynamic Profiles to Apply Services to DHCP Subscriber Interfaces

Dynamic Profile Attachment to DHCP Subscriber Interfaces Overview

The router's DHCP support enables you to attach a dynamic profile to a DHCP subscriber interface. When a DHCP subscriber logs in, the router instantiates the specified dynamic profile and then applies the services defined in the profile to the interface.

You can attach dynamic profiles to all interfaces or you can specify a particular group of interfaces to which the profile is attached. Both the DHCP local server and the DHCP relay agent support the attachment of dynamic profiles to interfaces.

You can enable the following optional features when the dynamic profile is attached. The two options cannot be used together.

- Enable multiple DHCP subscribers to share the same VLAN logical interface. The firewall filters, CoS schedulers, and IGMP configuration of the clients are merged.
- Specify the primary dynamic profile that is instantiated when the first subscriber logs in.

Multiple DHCP Subscribers Sharing the Same VLAN Logical Interface

The **aggregate-clients** statement specifies that the router merge the firewall filters, CoS schedulers, and IGMP configuration of multiple DHCP clients that are on the same VLAN logical interface (for example, multiple clients belonging to the same household). You can configure the aggregate-clients support for all interfaces or for a group of interfaces. The **aggregate-clients** statement provides the option of either merging (chaining) or replacing software components for each client.

By default, the feature is disabled and a single DHCP client is allowed per VLAN when a dynamic profile is associated with the VLAN logical interface.

When you specify the **merge** option, the router aggregates the software components for multiple subscribers as follows:

- Firewall filters—The filters are chained together using the precedence as the order of execution. If the same firewall filter is attached multiple times, the filter is executed only once.
- CoS schedulers—The different CoS schedulers are merged as if the scheduler map has multiple schedulers. The merge operation for the individual traffic-control-profiles parameters (shaping-rate, delay-buffer-rate, guaranteed-rate) preserves the maximum value for each parameter.
- IGMP configuration—The current IGMP configuration is replaced with the configuration of the newest DHCP client.

When you specify the **replace** option, the entire logical interface is replaced whenever a new client logs in to the network using the same VLAN logical interface. For example, if a customer subscribes to voice, video, and data services on the network, when a voice client logs in, instead of applying a specific voice filter for only that service, the entire voice, video, and data filter chain is applied.

NOTE: You cannot use a dynamic demux interface to represent multiple subscribers in a dynamic profile attached to an interface. One dynamic demux interface represents one subscriber. Do not configure the **aggregate-clients** option when attaching a dynamic profile to a demux interface for DHCP.

Primary Dynamic Profile

The **use-primary** option enables you to specify the primary dynamic profile that is instantiated when the first subscriber logs in. Subsequent subscribers are not assigned the primary dynamic profile; instead, they are assigned the dynamic profile specified for the interface. When the first subscriber logs out, the next subscriber that logs in is assigned the primary dynamic profile.

This feature can conserve logical interfaces in a network where dynamic IP demux interfaces are used to represent subscribers. To conserve interfaces, make sure the primary profile that you specify does not create a demux interface, but provides the initial policies for the primary interface subscriber.

Related Documentation

- Attaching Dynamic Profiles to DHCP Subscriber Interfaces or DHCP Client Interfaces on page 114

Attaching Dynamic Profiles to DHCP Subscriber Interfaces or DHCP Client Interfaces

This topic describes how to attach a dynamic profile to a DHCP subscriber interface or a DHCP client interface. When a DHCP subscriber or DHCP client logs in, the specified dynamic profile is instantiated and the services defined in the profile are applied to the interface.

This topic contains the following sections:

Attaching a Dynamic Profile to All DHCP Subscriber or All DHCP Client Interfaces

To attach a dynamic profile to all DHCP subscriber or all DHCP client interfaces:

1. At the DHCP configuration hierarchy, use the **dynamic-profile** statement to specify the name of the dynamic profile to attach to all interfaces.
   - For DHCP local server:

     [edit system services dhcp-local-server]
     user@host# **set dynamic-profile vod-profile-22**
   - For DHCP relay agent:

     [edit forwarding-options dhcp-relay]
     user@host# **set dynamic-profile vod-profile-west**
2. (Routers only) Optionally, you can configure the attribute to use when attaching the specified profile.

   You can include either the **aggregate-clients** option to enable multiple DHCP subscribers to share the same VLAN logical interface, or the **use-primary** option to specify that the primary dynamic profile is used. The **aggregate-clients** option does not apply to demux subscriber interfaces. The two options are mutually exclusive.
   - To enable multiple subscribers to share the same VLAN logical interface:

     [edit system services dhcp-local-server dynamic-profile]
     user@host# **set aggregate-clients merge**
   - To use the primary dynamic profile:

     [edit forwarding-options dhcp-relay dynamic-profile]
     user@host# **set use-primary subscriber_profile**

Attaching a Dynamic Profile to a Group of DHCP Subscriber Interfaces or a Group of DHCP Client Interfaces

Before you begin:

- Configure the interface group.

  See *Grouping Interfaces with Common DHCP Configurations*.

To attach a dynamic profile to a group of interfaces:

1. At the DHCP configuration hierarchy, specify the name of the interface group and the dynamic profile to attach to the group.

- For DHCP local server:

  [edit system services dhcp-local-server]
  user@host# **set group boston dynamic-profile vod-profile-42**

- For DHCP relay agent:

  [edit forwarding-options dhcp-relay]
  user@host# **set group quebec dynamic-profile vod-profile-east**

2. (Routers only) Optionally, you can configure the attribute to use when attaching the specified profile.

   You can include either the **aggregate-clients** option to enable multiple DHCP subscribers to share the same VLAN logical interface, or the **use-primary** option to specify that the primary dynamic profile is used. The **aggregate-clients** option does not apply to demux subscriber interfaces. The two options are mutually exclusive.

   - To enable multiple subscribers to share the same VLAN logical interface:

     [edit system services dhcp-local-server dynamic-profile]
     user@host# **set aggregate-clients merge**

   - To use the primary dynamic profile:

     [edit forwarding-options dhcp-relay dynamic-profile]
     user@host# **set use-primary subscriber_profile**

Related Documentation

- *Dynamic Profiles Overview*
- Dynamic Profile Attachment to DHCP Subscriber Interfaces Overview on page 113
- *Example: Configuring Dynamic Subscriber Interfaces on IP Demux Interfaces*

CHAPTER 12

Configuring DHCP IP Demux and PPPoE Demux Interfaces Over the Same VLAN

Example: Concurrent Configuration of Dynamic DHCP IP Demux and PPPoE Demux Interfaces over the Same VLAN Demux Interface

This example shows how to configure both dynamic DHCP IP demux and PPPoE demux interfaces over the same dynamic VLAN demux interface. The example provides an IPv4 configuration. However, you can also configure concurrent IP over Ethernet/DHCP and PPPoE interfaces over the same VLAN interface using IPv6 addressing.

NOTE: You can also configure dynamic IP over Ethernet/DHCP and PPPoE interfaces concurrently over the same static VLAN interface. For information on how to configure static VLAN interfaces, see *Configuring a Subscriber Interface with a Static VLAN Interface*.

Requirements

Before you begin, make sure to configure either DHCP Relay or DHCP Local Server. For information about configuring either of these components, see *Extended DHCP Relay Agent Overview* or *Extended DHCP Local Server Overview*.

Overview

With the introduction of the **family pppoe** statement, PPPoE is no longer treated as an exclusive encapsulation configuration and you can configure VLAN interfaces with multiple protocol interface stacks. For example, you can configure IP over Ethernet/DHCP and PPPoE interfaces concurrently over a single VLAN interface.

Configuration

Preparing a Subscriber Access Interface

CLI Quick Configuration

To quickly configure the aggregated Ethernet interface over which subscribers access the router:

```
[edit]
set chassis aggregated-devices ethernet device-count 1
set interfaces ge-5/0/9 gigether-options 802.3ad ae0
set interfaces ge-5/1/9 gigether-options 802.3ad ae0
set interfaces ae0 flexible-vlan-tagging
set interfaces ae0 auto-configure vlan-ranges dynamic-profile Auto-VLAN-Demux accept
  any
set interfaces ae0 auto-configure vlan-ranges dynamic-profile Auto-VLAN-Demux ranges
  ranges 1000-1500
set interfaces ae0 auto-configure stacked-vlan-ranges dynamic-profile
  Auto-Stacked-VLAN-Demux accept any
set interfaces ae0 auto-configure stacked-vlan-ranges dynamic-profile
  Auto-Stacked-VLAN-Demux ranges 1501-2000,any
set interfaces ae0 aggregated-ether-options lacp active
set interfaces ae0 aggregated-ether-options lacp link-protection
```

Step-by-Step Procedure

When configuring multiple protocol interface stacks concurrently over the same VLAN interface, you must configure physical interfaces over which DHCP or PPPoE clients initially access the router. We recommend that you specify the same VLAN tagging for the interface that you expect from incoming clients. This example uses flexible VLAN tagging to simultaneously support transmission of 802.1Q VLAN single-tag and dual-tag frames on logical interfaces on the same Ethernet port.

To automatically create dynamic VLANs, the interface must also include the VLAN range type (single or stacked), dynamic profile reference, and any specific ranges you want the VLANs to use.

To configure a physical interface for subscriber access:

1. Access the physical interface over which you want subscribers to initially access the router.

   ```
   [edit]
   user@host# edit interfaces ge-5/0/9
   ```

2. Specify the aggregated Ethernet interface to which the physical interface belongs.

   ```
   [edit interfaces ge-5/0/9]
   user@host# set gigether-options 802.3ad ae0
   ```


3. Repeat Step 1 and Step 2for each interface you want to assign to the aggregated Ethernet bundle.

   ```
   [edit]
   user@host# set interfaces ge-5/1/9 gigether-options 802.3ad ae0
   ```

4. Access the aggregated Ethernet interface.

   ```
   [edit]
   user@host# edit interfaces ae0
   ```

5. Specify the VLAN tagging that you want the aggregated Ethernet interfaces to use.

   ```
   [edit interfaces ae0]
   user@host# set flexible-vlan-tagging
   ```

6. Edit the **auto-configure** stanza to automatically configure VLANs.

   ```
   [edit interfaces ae0]
   user@host# edit auto-configure
   ```

7. Edit the **vlan-ranges** stanza for single-tagged VLANs.

   ```
   [edit interfaces ae0 auto-configure]
   user@host# edit vlan-ranges
   ```

8. Specify the dynamic VLAN profile that you want the interface to use for dynamically creating single-tagged VLANs.

   ```
   [edit interfaces ae0 auto-configure vlan-ranges]
   user@host# edit dynamic-profile Auto-VLAN-Demux
   ```

9. Specify what VLAN Ethernet packet type the VLAN profile accepts.

   ```
   [edit interfaces ae0 auto-configure vlan-ranges dynamic-profile Auto-VLAN-Demux]
   user@host# set accept any
   ```

10. Specify the VLAN ranges that you want the dynamic profile to use. The following example specifies a lower VLAN ID limit of 1000 and an upper VLAN ID limit of 1500.

    ```
    [edit interfaces ae0 auto-configure vlan-ranges dynamic-profile Auto-VLAN-Demux]
    user@host# set ranges 1000-1500
    ```

11. Edit the **stacked-vlan-ranges** stanza for the dual-tagged VLANs.

    ```
    [edit interfaces ae0 auto-configure]
    user@host# edit stacked-vlan-ranges
    ```

12. Specify the dynamic VLAN profile that you want the interface to use for dynamically creating dual-tagged VLANs.

    ```
    [edit interfaces ae0 auto-configure stacked-vlan-ranges]
    user@host# edit dynamic-profile Auto-Stacked-VLAN-Demux
    ```

13. Specify what VLAN Ethernet packet type the stacked VLAN profile accepts.

    ```
    [edit interfaces ae0 auto-configure stacked-vlan-ranges dynamic-profile
      Auto-Stacked-VLAN-Demux]
    user@host# set accept any
    ```

14. Specify the outer and inner stacked VLAN ranges that you want the dynamic profile to use. The following example specifies an outer stacked VLAN ID range from 1501 through 2000 (to avoid overlapping VLAN IDs with single-tag VLANs) and an inner

stacked VLAN ID range of any (enabling a range from 1 through 4094 for the inner stacked VLAN ID).

```
[edit interfaces ge-5/0/9 auto-configure stacked-vlan-ranges dynamic-profile
  Auto-Stacked-VLAN-Demux]
user@host# set ranges 1501-2000,any
```

15. (Optional) Activate the transmission of LACP packets on the aggregated Ethernet interfaces.

```
[edit interfaces ae0]
user@host# set aggregated-ether-options lacp active
```

16. Specify that the aggregated Ethernet interfaces use link protection.

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

Preparing the Loopback Interface

CLI Quick Configuration

To quickly configure the required loopback interface for this example:

```
[edit]
set interfaces lo0.0 unit 0 family inet address 100.100.100.1/32
```

Step-by-Step Procedure

You must configure a loopback interface for use as the unnumbered address and preferred source address for dynamically created interfaces.

To configure the required loopback interface for this example:

1. Configure a loopback interface.

```
[edit]
user@host# edit interfaces lo0.0
```

2. Specify that the loopback interface accept inet packets.

```
[edit interfaces lo0 unit 0]
user@host# edit family inet
```

3. Specify the IP address for the loopback interface.

```
[edit interfaces lo0 unit 0 family inet]
user@host# set address 100.100.100.1/32
```

Configuring a Dynamic Profile to Create Dynamic Single-Tagged VLANs

CLI Quick Configuration

To quickly configure the dynamic profile used to dynamically create single-tagged VLANs in the example:

```
[edit]
set dynamic-profiles Auto-VLAN-Demux interfaces demux0 unit $junos-interface-unit
  demux-source inet
set dynamic-profiles Auto-VLAN-Demux interfaces demux0 unit $junos-interface-unit
  proxy-arp
set dynamic-profiles Auto-VLAN-Demux interfaces demux0 unit $junos-interface-unit
  vlan-id $junos-vlan-id
```

```
set dynamic-profiles Auto-VLAN-Demux interfaces demux0 unit $junos-interface-unit
  demux options underlying-interface $junos-interface-ifd-name
set dynamic-profiles Auto-VLAN-Demux interfaces demux0 unit $junos-interface-unit
  family inet unnumbered-address lo0.0 preferred source-address 100.100.100.1
set dynamic-profiles Auto-VLAN-Demux interfaces demux0 unit $junos-interface-unit
  family pppoe duplicate-protection
set dynamic-profiles Auto-VLAN-Demux interfaces demux0 unit $junos-interface-unit
  family pppoe dynamic-profile PPP-Base-PAP
```

Step-by-Step Procedure

For both dynamic DHCP IP demux and dynamic PPPoE interfaces to reside concurrently on a single-tagged VLAN interface, the VLAN interface must first exist.

To configure a dynamic profile that automatically creates VLAN interfaces:

1. Create a dynamic profile for automatically creating VLAN interfaces.

   ```
   [edit]
   user@host# edit dynamic-profiles Auto-VLAN-Demux
   ```

2. Specify that the dynamic VLAN profile use the demux interface.

   ```
   [edit dynamic-profiles “Auto-VLAN-Demux”]
   user@host# edit interfaces demux0
   ```

3. Specify that the dynamic profile apply the demux interface unit value to the dynamic VLANs.

   ```
   [edit dynamic-profiles Auto-VLAN-Demux interfaces demux0]
   user@host# edit unit $junos-interface-unit
   ```

4. Specify that the demux source accept IPv4 (inet) packets.

   ```
   [edit dynamic-profiles Auto-VLAN-Demux interfaces demux0 unit
     “$junos-interface-unit”]
   user@host# set demux-source inet
   ```

5. (Optional) Specify that each dynamically created interface respond to any ARP request, as long as an active route exists to the target address of the ARP request.

   ```
   [edit dynamic-profiles Auto-VLAN-Demux interfaces demux0 unit
     “$junos-interface-unit”]
   user@host# set proxy-arp
   ```

6. Specify that VLAN IDs are dynamically created.

   ```
   [edit dynamic-profiles Auto-VLAN-Demux interfaces demux0 unit
     “$junos-interface-unit”]
   user@host# set vlan-id $junos-vlan-id
   ```

7. Specify the logical underlying interface for the dynamic VLANs.

   ```
   [edit dynamic-profiles Auto-VLAN-Demux interfaces demux0 unit
     “$junos-interface-unit”]
   user@host# set demux-options underlying-interface $junos-interface-ifd-name
   ```

8. Specify that the VLAN demux interface can accept inet family packets for IP over Ethernet/DHCP subscribers.

   ```
   [edit dynamic-profiles Auto-VLAN-Demux interfaces demux0 unit
     “$junos-interface-unit”]
   user@host# edit family inet
   ```


9. Specify the loopback address as the unnumbered address and preferred source address for the inet family.

```
[edit dynamic-profiles Auto-VLAN-Demux interfaces demux0 unit
  "$junos-interface-unit" family inet]
user@host# set unnumbered-address lo0.0 preferred-source-address 100.100.100.1
```

10. Specify that the VLAN demux interface can accept pppoe family packets for PPPoE subscribers.

```
[edit dynamic-profiles Auto-VLAN-Demux interfaces demux0 unit
  "$junos-interface-unit"]
user@host# edit family pppoe
```

11. Prevent multiple PPPoE sessions from being created for the same PPPoE subscriber on the same VLAN interface.

```
[edit dynamic-profiles Auto-VLAN-Demux interfaces demux0 unit
  "$junos-interface-unit" family pppoe]
user@host# set duplicate-protection
```

12. Apply the dynamic PPP interface profile to any dynamic PPP interfaces.

```
[edit dynamic-profiles Auto-VLAN-Demux interfaces demux0 unit
  "$junos-interface-unit" family pppoe]
user@host# set dynamic-profile PPP-Base-PAP
```

Configuring a Dynamic Profile to Create Dynamic Dual-Tagged VLANs

CLI Quick Configuration

To quickly configure the dynamic profile used to dynamically create stacked/dual-tagged VLANs in the example:

```
[edit]
set dynamic-profiles Auto-Stacked-VLAN-Demux interfaces demux0 unit
  $junos-interface-unit demux-source inet
set dynamic-profiles Auto-Stacked-VLAN-Demux interfaces demux0 unit
  $junos-interface-unit proxy-arp
set dynamic-profiles Auto-Stacked-VLAN-Demux interfaces demux0 unit
  $junos-interface-unit vlan-tags outer $junos-stacked-vlan-id
set dynamic-profiles Auto-Stacked-VLAN-Demux interfaces demux0 unit
  $junos-interface-unit vlan-tags inner $junos-vlan-id
set dynamic-profiles Auto-Stacked-VLAN-Demux interfaces demux0 unit
  $junos-interface-unit demux options underlying-interface $junos-interface-ifd-name
set dynamic-profiles Auto-Stacked-VLAN-Demux interfaces demux0 unit
  $junos-interface-unit family inet unnumbered-address lo0.0 preferred source-address
  100.100.100.1
set dynamic-profiles Auto-Stacked-VLAN-Demux interfaces demux0 unit
  $junos-interface-unit family pppoe duplicate-protection
set dynamic-profiles Auto-Stacked-VLAN-Demux interfaces demux0 unit
  $junos-interface-unit family pppoe dynamic-profile PPP-Base-PAP
```


Step-by-Step Procedure

For both dynamic DHCP IP demux and dynamic PPPoE interfaces to reside concurrently on a VLAN interface, the VLAN interface must first exist.

To configure a dynamic profile that automatically creates stacked/dual-tagged VLAN interfaces:

1. Create a dynamic profile for automatically creating VLAN interfaces.

   ```
   [edit]
   user@host# edit dynamic-profiles Auto-Stacked-VLAN-Demux
   ```

2. Specify that the dynamic VLAN profile use the demux interface.

   ```
   [edit dynamic-profiles “Auto-Stacked-VLAN-Demux”]
   user@host# edit interfaces demux0
   ```

3. Specify that the dynamic profile apply the demux interface unit value to the dynamic VLANs.

   ```
   [edit dynamic-profiles Auto-Stacked-VLAN-Demux interfaces demux0]
   user@host# edit unit $junos-interface-unit
   ```

4. Specify that the demux source accept IPv4 (inet) packets.

   ```
   [edit dynamic-profiles Auto-Stacked-VLAN-Demux interfaces demux0 unit
     “$junos-interface-unit”]
   user@host# set demux-source inet
   ```

5. (Optional) Specify that each dynamically created interface respond to any ARP request, as long as an active route exists to the target address of the ARP request.

   ```
   [edit dynamic-profiles Auto-Stacked-VLAN-Demux interfaces demux0 unit
     “$junos-interface-unit”]
   user@host# set proxy-arp
   ```

6. Specify that the outer VLAN ID is dynamically created.

   ```
   [edit dynamic-profiles Auto-Stacked-VLAN-Demux interfaces demux0 unit
     “$junos-interface-unit”]
   user@host# set vlan-id -tags outer $junos-stacked-vlan-id
   ```

7. Specify that the inner VLAN ID is dynamically created.

   ```
   [edit dynamic-profiles Auto-Stacked-VLAN-Demux interfaces demux0 unit
     “$junos-interface-unit”]
   user@host# set vlan-id -tags inner $junos-vlan-id
   ```

8. Specify the logical underlying interface for the dynamic VLANs.

   ```
   [edit dynamic-profiles Auto-Stacked-VLAN-Demux interfaces demux0 unit
     “$junos-interface-unit”]
   user@host# set demux-options underlying-interface $junos-interface-ifd-name
   ```

9. Specify that the VLAN demux interface can accept inet family packets for IP over Ethernet/DHCP subscribers.

   ```
   [edit dynamic-profiles Auto-Stacked-VLAN-Demux interfaces demux0 unit
     “$junos-interface-unit”]
   user@host# edit family inet
   ```

10. Specify the loopback address as the unnumbered address and preferred source address for the inet family.


```
[edit dynamic-profiles Auto-Stacked-VLAN-Demux interfaces demux0 unit
  "$junos-interface-unit" family inet]
user@host# set unnumbered-address lo0.0 preferred-source-address 100.100.100.1
```

11. Specify that the VLAN demux interface can accept pppoe family packets for PPPoE subscribers.

```
[edit dynamic-profiles Auto-Stacked-VLAN-Demux interfaces demux0 unit
  "$junos-interface-unit"]
user@host# edit family pppoe
```

12. Prevent the activation of another dynamic PPPoE logical interface on the same demux underlying interface.

```
[edit dynamic-profiles Auto-Stacked-VLAN-Demux interfaces demux0 unit
  "$junos-interface-unit" family pppoe]
user@host# set duplicate-protection
```

13. Apply the dynamic PPP interface profile to any dynamic PPP interfaces.

```
[edit dynamic-profiles Auto-Stacked-VLAN-Demux interfaces demux0 unit
  "$junos-interface-unit" family pppoe]
user@host# set dynamic-profile PPP-Base-PAP
```

Configuring a Dynamic Profile to Create Dynamic IP Demux Interfaces

CLI Quick Configuration

To quickly configure the dynamic profile used to dynamically create DHCP IP demux interfaces in the example:

```
[edit]
set dynamic-profiles DHCP-IP-Demux interfaces demux0 unit $junos-interface-unit
  proxy-arp
set dynamic-profiles DHCP-IP-Demux interfaces demux0 unit $junos-interface-unit
  demux-options underlying-interface $junos-underlying-interface
set dynamic-profiles DHCP-IP-Demux interfaces demux0 unit $junos-interface-unit
  family inet demux-source $junos-subscriber-ip-address
set dynamic-profiles DHCP-IP-Demux interfaces demux0 unit $junos-interface-unit
  family inet unnumbered-address lo0.0 preferred-source-address 100.100.100.1
```

Step-by-Step Procedure

To configure a dynamic profile that automatically creates IP demux interfaces:

1. Create a dynamic profile for dynamically creating IP demux interfaces.

```
[edit]
user@host# edit dynamic-profiles DHCP-IP-Demux
```

2. Specify that the dynamic profile use the demux0 interface.

```
[edit dynamic-profiles DHCP-IP-Demux]
user@host# edit interfaces demux0
```

3. Specify that the dynamic profile apply the interface unit value to the dynamic PPPoE interfaces.

```
[edit dynamic-profiles DHCP-IP-Demux interfaces demux0]
user@host# edit unit $junos-interface-unit
```

4. (Optional) Configure the router to respond to any ARP request, as long as the router has an active route to the target address of the ARP request.


```
[edit dynamic-profiles DHCP-IP-Demux interfaces demux0 unit
  “$junos-interface-unit”]
user@host# set proxy-arp
```

5. Specify the logical underlying interface for the dynamic IP demux interfaces.

```
[edit dynamic-profiles DHCP-IP-Demux interfaces demux0 unit
  “$junos-interface-unit”]
user@host# set demux-options underlying-interface $junos-underlying-interface
```

6. Specify the protocol family information for the dynamic IP demux interfaces.

```
[edit dynamic-profiles DHCP-IP-Demux interfaces demux0 unit
  “$junos-interface-unit”]
user@host# edit family inet
```

7. Specify the demux source address is obtained from the incoming subscriber IP address.

```
[edit dynamic-profiles DHCP-IP-Demux interfaces demux0 unit “$junos-interface-unit”
  family inet]
user@host# set demux-source $junos-subscriber-ip-address
```

8. Specify the loopback interface as the unnumbered address and the demux interface IP address as the preferred source address for the dynamic IP demux interfaces.

```
[edit dynamic-profiles DHCP-IP-Demux interfaces demux0 unit “$junos-interface-unit”
  family inet]
user@host# set unnumbered-address lo0.0 preferred-source-address 100.100.100.1
```

Configuring a Dynamic Profile to Create Dynamic PPPoE Interfaces

CLI Quick Configuration

To quickly configure the dynamic profile used to dynamically create PPPoE interfaces in the example:

```
[edit]
set dynamic-profiles PPP-Base-PAP interfaces pp0 unit $junos-interface-unit ppp-options
  pap
set dynamic-profiles PPP-Base-PAP interfaces pp0 unit $junos-interface-unit
  pppoe-options underlying-interface $junos-underlying-interface server
set dynamic-profiles PPP-Base-PAP interfaces pp0 unit $junos-interface-unit
  no-keepalives
set dynamic-profiles PPP-Base-PAP interfaces pp0 unit $junos-interface-unit family inet
  unnumbered-address lo0.0
```

Step-by-Step Procedure

1. Create a dynamic profile for automatically creating PPPoE interfaces.

```
[edit]
user@host# edit dynamic-profiles PPP-Base-PAP
```

2. Specify that the dynamic PPPoE profile use the pp0 interface.

```
[edit dynamic-profiles PPP-Base-PAP]
user@host# edit interfaces pp0
```

3. Specify that the dynamic profile apply the interface unit value to the dynamic PPPoE interfaces.

```
[edit dynamic-profiles PPP-Base-PAP interfaces pp0]
```

user@host# **edit unit $junos-interface-unit**

4. Specify that dynamically created PPPoE interfaces use PAP authentication.

   [edit dynamic-profiles PPP-Base-PAP interfaces pp0 unit "$junos-interface-unit"]
   user@host# **set ppp-options pap**

5. Specify the logical underlying interface for the dynamic PPPoE interfaces.

   [edit dynamic-profiles PPP-Base-PAP interfaces pp0 unit "$junos-interface-unit"]
   user@host# **set pppoe-options underlying-interface $junos-underlying-interface**

6. Specify that the router act as a PPPoE server.

   [edit dynamic-profiles PPP-Base-PAP interfaces pp0 unit "$junos-interface-unit"]
   user@host# **set pppoe-options server**

7. (Optional) Disable the sending of keepalive messages on the dynamic PPPoE interfaces.

   [edit dynamic-profiles PPP-Base-PAP interfaces pp0 unit "$junos-interface-unit"]
   user@host# **set no-keepalives**

8. Specify the protocol family information for the dynamic PPPoE interfaces.

   [edit dynamic-profiles PPP-Base-PAP interfaces pp0 unit "$junos-interface-unit"]
   user@host# **edit family inet**

9. Specify the loopback interface as the unnumbered address for the dynamic PPPoE interfaces.

   [edit dynamic-profiles PPP-Base-PAP interfaces pp0 unit "$junos-interface-unit"]
   user@host# **set unnumbered-address lo0.0**

Verification

- Subscriber Verification on page 126
- Interface Verification on page 126

Subscriber Verification

Purpose View subscriber information on the router.

Action

- To display dynamic subscriber information:

  user@host# **show subscribers detail**

Interface Verification

Purpose View interface-specific information on the router.

Action

- To display interface-specific output:

  user@host# **show interfaces *interface-name***

Related Documentation

- *Dynamic Profiles Overview*
- *Configuring a Basic Dynamic Profile*

- *Configuring Predefined Dynamic Variables in Dynamic Profiles*
- Dynamic 802.1Q VLAN Overview on page 5
- *Configuring VLAN Dynamic Profiles*
- *Demultiplexing Interface Overview*
- Configuring the PPPoE Family for an Underlying Interface on page 159

CHAPTER 13

Providing Security for DHCP Interfaces Using MAC Address Validation

MAC Address Validation for Subscriber Interfaces Overview

MAC address validation enables the router to validate that received packets contain a trusted IP source and an Ethernet MAC source address.

Configuring MAC address validation can provide additional validation when subscribers access billable services. MAC address validation provides additional security by enabling the router to drop packets that do not match, such as packets with spoofed addresses.

When subscribers log in, they are automatically assigned IP addresses by DHCP. With MAC address validation enabled, the router compares the IP source and MAC source addresses against trusted addresses, and forwards or drops the packets according to the match and the validation mode.

Supported Types of Subscriber Interfaces

MAC address validation is supported on statically or dynamically created Ethernet interfaces and demux interfaces on MX Series 3D Universal Edge Routers as follows:

- When the router is configured for a normal (non-enhanced) network services mode, MAC address validation is supported on both DPCs and MPCs. The router can be populated completely with one or the other type of line card, or have a mix of both types. Normal network services mode is the default.
- When the router is configured for Enhanced IP Network Services mode or Enhanced Ethernet Network Services mode, MAC address validation is supported only on MPCs. If the router has both DPCs and MPCs, or only DPCs, you cannot configure the chassis to be in enhanced mode.

MAC address validation is optimized for scaling when the router is in enhanced network services modes. Enhanced network services modes affect other features, such as multicast and firewall filters, so you must take that in to consideration when deciding whether to configure enhanced mode. For more information about the enhanced network service modes, see *Network Services Mode Overview*.

In normal network services mode, you can use the **show interfaces statistics** ***interface-name*** command to display a per-interface count of the packets that failed validation and were dropped. In enhanced network services mode, this command does not count the dropped packets; you must contact Juniper Networks Customer Support for assistance in collecting this data.

Trusted Addresses

A trusted address tuple is a 32–bit IP address and a 48–bit MAC address. Prefixes and ranges are not supported.

The IP source address and the MAC source address used for validation must be from a trusted source.

All static ARP addresses configured through the CLI are trusted addresses; dynamic ARP addresses are not considered trusted addresses.

Addresses dynamically created through an extended DHCP local server or extended DHCP relay are also trusted addresses. When a DHCP server and client negotiate an IP address, the resulting IP address and MAC address tuple is trusted. Each DHCP subscriber can generate more than one address tuple.

Each MAC address can have more than one IP address, which can result in more than one valid tuple. Each IP address must map to one MAC address.

Types of MAC Address Validation

You can configure either of two types or modes of MAC address validation, loose or strict. The behavior of the two modes varies depending on how well the incoming packets match the trusted address tuples. The modes differ only when the IP source address alone does not match any trusted IP address. Table 5 on page 130 compares the behavior of the two modes. Dropped packets are considered to be spoofed.

Table 5: Comparison of MAC Address Validation Modes

Incoming Packet Addresses Match Trusted Address Tuple	Loose Mode Action	Strict Mode Action
• IP source address matches and • MAC source address matches	Forwards packet	Forwards packet
• IP source address matches but • MAC source address does not match	Drops packet	Drops packet
• IP source address does not match and • MAC source address either matches or does not match	Forwards packet	Drops packet

Configuring strict mode is a more conservative strategy because it requires both received source addresses to match trusted addresses.

When you configure MAC address validation for IP demux interfaces in a dynamic profile and specify either loose or strict validation, the resulting behavior is always loose validation. To enable strict behavior for a dynamic IP demux interface, you must configure strict validation for both the IP demux interface and the underlying interface.

Related Documentation

- Configuring MAC Address Validation for Subscriber Interfaces on page 131

Configuring MAC Address Validation for Subscriber Interfaces

This topic describes how to configure MAC address validation for subscriber interfaces in dynamic profiles on MX Series routers.

The subscriber interfaces can be statically created and associated with a dynamic profile (for example, VLAN interfaces) or dynamically created in the dynamic profile (such as demux interfaces).

By default, MAC address validation is disabled.

This topic contains the following sections:

Configuring MAC Address Validation for Static Subscriber Interfaces

This topic describes how to configure MAC address validation for static subscriber interfaces in dynamic profiles on MX Series routers.

Before you begin:

- Configure the dynamic profile.

  See *Configuring a Basic Dynamic Profile*.

- (Optional) Configure an enhanced network services mode.

  See *Configuring Junos OS to Run a Specific Network Services Mode in MX Series Routers*.

To configure MAC address validation on static subscriber interfaces:

1. Configure the static VLAN interface.

   [edit interfaces]
   user@host# **set** ***interface-name*** **unit** ***logical-unit-number*** **family inet**

2. Configure the type of MAC address validation for the interface.

   - To configure loose validation:

     [edit interfaces *interface-name* unit *logical-unit-number* family inet]
     user@host# **set mac-validate loose**

- To configure strict validation:

  [edit interfaces *interface-name* unit *logical-unit-number* family inet]
  user@host# **set** mac-validate **strict**

For example, to configure loose validation on interface fe-0/0/0.0, configure the following:

[edit interfaces fe-0/0/0 unit 0 family inet]
user@host# **set** mac-validate **loose**

After you configure MAC address validation:

- Associate the static VLAN interface with the dynamic profile.

  See *Associating Dynamic Profiles with Statically Created Interfaces*.

Configuring MAC Address Validation for Dynamic Subscriber Interfaces

This topic describes how to configure MAC address validation for subscriber interfaces created on demux interfaces in dynamic profiles on MX Series routers.

When you configure MAC address validation for demux interfaces in a dynamic profile and specify either **loose** or **strict** validation, the resulting behavior is always loose validation. To enable strict behavior for a dynamic IP demux interface, besides configuring either **loose** or **strict** mode on the IP demux interface, you must also configure strict validation on the underlying interface.

Before you begin:

- Configure the dynamic profile.

  See *Configuring a Basic Dynamic Profile*.

- Configure the dynamic IP demux interface.

  See "Configuring Dynamic Subscriber Interfaces Using IP Demux Interfaces in Dynamic Profiles" on page 77.

- (Optional) Configure an enhanced network services mode.

  See *Configuring Junos OS to Run a Specific Network Services Mode in MX Series Routers*.

To configure loose MAC address validation for a dynamic subscriber interface:

- Configure loose validation for the demux interface.

  [edit dynamic-profiles *profile-name* interfaces demux0 unit "$junos-interface-unit"
    family inet]
  user@host# **set** mac-validate **loose**

  For loose validation, you do not need to configure MAC address validation on the underlying interface.

To configure strict MAC address validation for a dynamic subscriber interface:

1. Configure validation for the demux interface.

```
[edit dynamic-profiles profile-name interfaces demux0 unit "$junos-interface-unit"
   family inet]
user@host# set mac-validate validation-mode
```

NOTE: Remember, although you must configure validation on the IP demux interface, it does not matter which mode you specify because the behavior is always loose.

2. Configure strict validation for the underlying interface.

```
[edit interfaces interface-name unit logical-unit-number family inet]
user@host# set mac-validate strict
```

The underlying interface in this case is statically configured—for example, ge-1/0/0.1—and assigned to a DHCP configuration group that is associated with the dynamic profile. In a more complicated configuration, the underlying interface itself can be configured by a dynamic profile; in that case the validation is configured in the profile that creates the underlying interface.

Related Documentation

- MAC Address Validation for Subscriber Interfaces Overview on page 129
- *Example: Configuring Dynamic Subscriber Interfaces on IP Demux Interfaces*

CHAPTER 14

Verifying Configuration and Status of Dynamic Subscribers

Verifying Configuration and Status of Dynamic Subscribers and Associated Sessions, Services, and Firewall Filters

Purpose

Verify configuration and status of dynamic subscribers, sessions, services, and firewall filters.

You can display information about subscribers in different ways, depending on the options you use with the **show subscriber** command. You can use details from one set of output with another command to display more detailed information of interest.

Action

- To display basic information for all subscribers:

```
user@host> show subscribers
Interface IP Address/VLAN ID User Name LS:RI
demux0.1073741824 0x8100.1500 0x8100.2900 test1@test.com default:testnet
demux0.1073741825 0x8100.1500 0x8100.2901 test1@test.com default:testnet
demux0.1073741826 0x8100.1500 0x8100.2902 test1@test.com default:testnet
demux0.1073741827 0x8100.1500 0x8100.2903 test1@test.com default:testnet
demux0.1073741826 172.16.200.6 test1@test.com default:testnet
demux0.1073741827 172.16.200.7 test1@test.com default:testnet
demux0.1073741824 172.16.200.8 test1@test.com default:testnet
demux0.1073741825 172.16.200.9 test1@test.com default:testnet
demux0.1073741828 0x8100.1500 0x8100.2910 test1@test.com default:default
demux0.1073741828 20.20.0.2 test1@test.com default:default
```

- To display more detailed information about a particular subscriber interface:

```
user@host> show subscribers interface demux0.1073741826 extensive
Type: VLAN
User Name: test1@test.com
Logical System: default
Routing Instance: testnet
Interface: demux0.1073741826
Interface type: Dynamic
Dynamic Profile Name: profile-vdemux-relay-23qos
MAC Address: 00:00:6e:56:01:04
State: Active
Radius Accounting ID: 12
Session ID: 12
```

```
Stacked VLAN Id: 0x8100.1500
VLAN Id: 0x8100.2902
Login Time: 2011-10-20 16:21:59 EST

Type: DHCP
User Name: test1@test.com
IP Address: 172.16.200.6
IP Netmask: 255.255.255.0
Logical System: default
Routing Instance: testnet
Interface: demux0.1073741826
Interface type: Static
MAC Address: 00:00:6e:56:01:04
State: Active
Radius Accounting ID: 21
Session ID: 21
Login Time: 2011-10-20 16:24:33 EST
Service Sessions: 2

Service Session ID: 25
Service Session Name: SUB-QOS
State: Active

Service Session ID: 26
Service Session Name: service-cb-content
State: Active
IPv4 Input Filter Name: content-cb-in-demux0.1073741826-in
IPv4 Output Filter Name: content-cb-out-demux0.1073741826-out
```

- To display traffic information for firewall filters.

```
user@host> show firewall
...
Filter: content-cb-in-demux0.1073741826-in
Counters:
Name      Bytes   Packets
__junos-dyn-service-counter  84336      1004

Filter: content-cb-out-demux0.1073741826-out
Counters:
Name      Bytes    Packets
__junos-dyn-service-counter      0       0
...
```

Instead of issuing successive commands to track the details for one subscriber interface, you can choose to display detailed information for all subscribers. However, the more subscribers you have, the more tedious it becomes to look through all the results for particular items of interest.

- To display detailed information for all subscribers:

  user@host> show subscribers **detail**

  user@host> show subscribers **extensive**

Meaning The output examples in this section show increasingly detailed information about dynamically created subscriber interfaces, including how many there are, what they are, and their characteristics; how many service sessions are active and what they are; whether firewall filters are attached to the sessions and what those filters are; and how much, if any, traffic is being filtered.

In the sample output shown here, the **show subscriber** command lists all the subscriber logical interfaces, including demux0.1073741826. You then display details about that interface and its associated subscribers with the **show subscribers interface demux0.1073741826 extensive** command. The Service Session Name fields for service sessions 25 and 26 in that output show two services are active on the interface, SUB-QOS and service-cb-content. The IPv4 Input Filter Name and the IPv4 Output Filter Name fields show that two filters have been applied to the service-cb-content session: content-cb-in-demux0.1073741826-in and content-cb-out-demux0.1073741826-out. You then use the **show firewalls** command to list the filters and see how much, if any, traffic is being filtered.

Related Documentation

- CLI Explorer

PART 3

Configuring PPPoE Subscriber Interfaces

CHAPTER 15

Configuring Dynamic PPPoE Subscriber Interfaces

Subscriber Interfaces and PPPoE Overview

You can configure the router to dynamically create Point-to-Point Protocol over Ethernet (PPPoE) logical interfaces on statically created underlying Ethernet interfaces. The router automatically and transparently creates the dynamic interface in response to the receipt of a PPPoE Active Discovery Request (PADR) control packet on the underlying interface. Because the router creates a dynamic PPPoE logical interface on demand when a subscriber logs in to the network, dynamic PPPoE logical interfaces are also referred to as *dynamic PPPoE subscriber interfaces*.

To enable the router to create a dynamic PPPoE logical interface on an underlying Ethernet interface, you define the attributes of the PPPoE logical interface in a dynamic profile, and then attach the dynamic profile to an Ethernet interface configured with PPPoE encapsulation. When the router receives a PADR control packet from a PPPoE client on an underlying interface with a PPPoE dynamic profile attached, the router uses the attributes defined in the profile to instantiate a dynamic PPPoE subscriber interface to handle the PPPoE session.

This overview covers the following topics:

- Benefits of Using Dynamic PPPoE Subscriber Interfaces on page 142
- Supported Platforms for Dynamic PPPoE Subscriber Interfaces on page 143
- Sequence of Operations for PPPoE Subscriber Access on page 143

Benefits of Using Dynamic PPPoE Subscriber Interfaces

Configuring and using dynamic PPPoE subscriber interfaces offers the following benefits:

- On-demand dynamic interface creation

  Configuring dynamic PPPoE subscriber interfaces provides the flexibility of dynamically creating the PPPoE subscriber interface only when needed; that is, when a subscriber logs in on the associated underlying Ethernet interface. By contrast, statically created interfaces allocate and consume system resources when the interface is created. Configuring and using dynamically created interfaces helps you effectively and conveniently manage edge or access networks in which large numbers of subscribers are constantly logging in to and logging out from the network on a transient basis.

- Dynamic removal of PPPoE subscriber interfaces without manual intervention

  When the PPPoE subscriber logs out or the PPPoE session is terminated, the router dynamically deletes the associated PPPoE subscriber interface without your intervention, thereby restoring any consumed resources to the router.

- Use of dynamic profiles to efficiently manage multiple subscriber interfaces

  A *dynamic profile* is a set of characteristics that can be dynamically assigned to subscriber interfaces. By using a profile, you reduce the management of a large number of interfaces by applying a set of common characteristics to multiple interfaces. When you configure a dynamic profile for PPPoE, you use predefined dynamic variables in the profile to represent information that varies from subscriber to subscriber, such as the logical unit number and underlying interface name. These variables are dynamically replaced with the values supplied by the network when the subscriber logs in.

- Denial of service (DoS) protection

  You can optionally configure the underlying Ethernet interface with certain PPPoE-specific attributes that can reduce the potential for DoS attacks. Duplicate protection, which is disabled by default, prevents activation of another dynamic PPPoE logical interface on the underlying interface when a PPPoE logical interface for the same client is already active on the underlying interface. You can also specify the maximum number of PPPoE sessions that the router can activate on the underlying interface. By enabling duplicate protection and restricting the maximum number of PPPoE sessions on the underlying interface, you can ensure that a single toxic PPPoE client cannot monopolize allocation of the PPPoE session.

- Support for dynamic PPPoE subscriber interface creation from PPPoE service name tables

  You can assign a previously configured PPPoE dynamic profile to a named, **empty**, or **any** service entry in a PPPoE service name table, or to an agent circuit identifier/agent remote identifier (ACI/ARI) pair defined for these services. The router uses the attributes defined in the profile to instantiate a dynamic PPPoE subscriber interface based on the service name, ACI, and ARI information provided by the PPPoE client during PPPoE negotiation. To specify the routing instance in which to instantiate the dynamic PPPoE subscriber interface, you can assign a previously configured routing instance to a named, **empty**, or **any** service, or to an ACI/ARI pair defined for these services. The dynamic profile and routing instance configured for the PPPoE service name table overrides the dynamic profile and routing instance assigned to the PPPoE underlying interface on which the dynamic subscriber interface is created.

Supported Platforms for Dynamic PPPoE Subscriber Interfaces

Configuration of dynamic PPPoE subscriber interfaces over static underlying Ethernet interfaces is supported on the following routing platforms:

- Intelligent Queuing 2 (IQ2) PICs on M120 Multiservice Edge Router and M320 Multiservice Edge Router
- MPC/MIC interfaces on MX Series 3D Universal Edge Routers

Sequence of Operations for PPPoE Subscriber Access

When a PPPoE subscriber logs in to the network, the PPPoE protocol defines the sequence of operations by which a connection is established and traffic flow is enabled on the dynamic PPPoE subscriber interface. Similarly, when the PPPoE subscriber logs out from the network, PPPoE defines the sequence that occurs to terminate the connection and remove the dynamic PPPoE subscriber interface from the router.

The router creates a dynamic PPPoE subscriber interface for each new PPPoE session, and removes the dynamic PPPoE subscriber interface when the session is terminated due to subscriber logout, PPP negotiation failure, or down status of the underlying Ethernet interface. Dynamic PPPoE subscriber interfaces are never reused for multiple PPPoE sessions.

Sequence When a PPPoE Subscriber Logs In

In a PPPoE subscriber network, the router acts as a *remote access concentrator*, also known as a *PPPoE server*. For a PPPoE client to initiate a PPPoE session with a PPPoE server, it must first perform PPPoE Discovery to identify the Ethernet MAC address of the remote access concentrator that can service its request. Based on the network topology, there may be more than one remote access concentrator with which the client can communicate. The Discovery process enables a PPPoE client to find all remote access concentrators and then select one to connect to.

The following sequence occurs when a PPPoE subscriber logs in to the network. Steps 1 through 5 in this sequence are part of the PPPoE Discovery process.

1. The PPPoE client broadcasts a PPPoE Active Discovery Initiation (PADI) packet to all remote access concentrators in the network.
2. One or more remote access concentrators respond to the PADI packet by sending a PPPoE Active Discovery Offer (PADO) packet, indicating that they can service the client request. The PADO packet includes the name of the access concentrator from which it was sent.
3. The client sends a unicast PPPoE Active Discovery Request (PADR) packet to the access concentrator it selects.

4. On receipt of the PADR packet on the underlying interface associated with a PPPoE dynamic profile, the router uses the attributes configured in the dynamic profile to create the dynamic PPPoE logical interface.
5. The router sends a PPPoE Active Discovery Session (PADS) packet to confirm establishment of the PPPoE connection.
6. The PPP Link Control Protocol (LCP) negotiates the PPP link between the client and the PPPoE server.
7. The subscriber is authenticated using the PPP authentication protocol (CHAP or PAP) configured in the PPPoE dynamic profile.
8. The PPP Network Control Protocol (NCP) negotiates the IP routing protocol and network family.
9. The PPP server issues an IP access address for the client, and the router adds the client access route to its routing table.
10. The router instantiates the dynamic profile and applies the attributes configured in the profile to the dynamic PPPoE subscriber interface.
11. PPP NCP negotiation completes, enabling traffic flow between the PPPoE client and the PPPoE server.

Sequence When a PPPoE Subscriber Logs Out

The following sequence occurs when a PPPoE subscriber logs out of the network:

1. The client terminates the PPP connection and the router receives an LCP termination request.
2. The router removes the client access router from its routing table.
3. The router sends or receives a PPPoE Active Discovery Termination (PADT) packet to end the PPPoE connection.
4. The router deactivates the subscriber, gathers final statistics for the PPPoE session, and sends the RADIUS server an Acct-Stop accounting message.
5. The router de-instantiates the PPPoE dynamic profile and removes the PPPoE logical interface. The router does not reuse the PPPoE logical interface for future dynamic PPPoE sessions.

Related Documentation

- Dynamic PPPoE Subscriber Interfaces over Static Underlying Interfaces Overview on page 153
- Configuring Dynamic PPPoE Subscriber Interfaces Using Dynamic Profiles on page 145

- *Configuring PPPoE Service Name Tables*

Configuring Dynamic PPPoE Subscriber Interfaces Using Dynamic Profiles

You can configure dynamic PPP-over-Ethernet (PPPoE) subscriber interfaces by using dynamic profiles. To enable the router to create a dynamic PPPoE subscriber interface on a PPPoE underlying interface, you define the attributes of the PPPoE logical interface in a dynamic profile, and then configure the underlying interface to use the dynamic profile.

To configure a dynamic PPPoE subscriber interface:

1. Configure a dynamic profile to define the attributes of the PPPoE logical interface.
   - To configure a basic dynamic profile for PPPoE subscriber access, see "Configuring a Basic PPPoE Dynamic Profile" on page 146.
   - To configure a dynamic profile for PPPoE with additional options for subscriber access, see "Configuring a PPPoE Dynamic Profile with Additional Options" on page 148.
2. Configure the underlying Ethernet interface to use the dynamic profile for PPPoE.

   See "Configuring an Underlying Interface for Dynamic PPPoE Subscriber Interfaces" on page 156.
3. (Optional) Assign a dynamic profile and routing instance to a service name or ACI/ARI pair in a PPPoE service name table to instantiate a dynamic PPPoE subscriber interface based on the information provided by the PPPoE client.

   See "Assigning a Dynamic Profile and Routing Instance to a Service Name or ACI/ARI Pair for Dynamic PPPoE Interface Creation" on page 150.
4. (Optional) Verify the dynamic PPPoE configuration by displaying or clearing PPPoE session statistics, and displaying information about the underlying Ethernet interface and PPPoE logical interface.

   See "Verifying and Managing Dynamic PPPoE Configuration" on page 211.

Related Documentation

- Subscriber Interfaces and PPPoE Overview on page 141
- Dynamic PPPoE Subscriber Interfaces over Static Underlying Interfaces Overview on page 153
- Example: Configuring a Dynamic PPPoE Subscriber Interface on a Static Gigabit Ethernet VLAN Interface on page 162
- Example: Configuring a PPPoE Service Name Table for Dynamic Subscriber Interface Creation on page 207

Configuring a Basic PPPoE Dynamic Profile

You can configure a basic dynamic profile for PPPoE subscribers that access the network. The dynamic profile defines the attributes of the dynamic PPPoE logical interface, also referred to as a *dynamic PPPoE subscriber interface*.

To provide basic access for PPPoE subscribers, the dynamic profile must provide a minimal configuration for a **pp0** (PPPoE) logical interface that includes the following:

- The logical unit number, represented by the **$junos-interface-unit** predefined dynamic variable
- The name of the underlying Ethernet interface, represented by the **$junos-underlying-interface** predefined dynamic variable
- The **server** statement, which configures the router to act as a PPPoE server
- The PPP authentication protocol (PAP or CHAP)
- The unnumbered address for the **inet** (IPv4) or **inet6** (IPv6) protocol family

NOTE: **The creation of dynamic PPPoE subscriber interfaces is supported for the inet and inet6 protocol families.**

To configure a basic PPPoE dynamic profile:

1. Name the dynamic profile.

   ```
   [edit]
   user@host# edit dynamic-profiles basic-pppoe-profile
   ```

2. Specify that you want to configure the **pp0** logical interface in the dynamic profile.

   ```
   [edit dynamic-profiles basic-pppoe-profile]
   user@host# edit interfaces pp0
   ```

3. Configure the predefined variable to represent the logical unit number for the **pp0** interface.

   You must use the **$junos-interface-unit** variable instead of the logical unit number for the **unit** statement. The **$junos-interface-unit** variable is dynamically replaced with the actual unit number supplied by the network when the subscriber logs in.

   ```
   [edit dynamic-profiles basic-pppoe-profile interfaces pp0]
   user@host# edit unit $junos-interface-unit
   ```

4. Configure PPPoE-specific options for the **pp0** interface.

   a. Configure the predefined variable to represent the name of the underlying Ethernet interface on which the router creates the dynamic PPPoE logical interface.

      You must use the **$junos-underlying-interface** variable instead of the underlying interface name for the **underlying-interface** statement. The **$junos-underlying-interface** variable is dynamically replaced with the actual name of the underlying interface supplied by the network when the subscriber logs in.


```
[edit dynamic-profiles basic-pppoe-profile interfaces pp0 unit
  "$junos-interface-unit"]
user@host# set pppoe-options underlying-interface $junos-underlying-interface
```

b. Configure the router to act as a PPPoE server, also known as a remote access concentrator.

```
[edit dynamic-profiles basic-pppoe-profile interfaces pp0 unit
  "$junos-interface-unit"]
user@host# set pppoe-options server
```

5. Configure the PPP authentication protocol for the **pp0** interface.

   For dynamic interfaces, the router supports only unidirectional authentication; that is, the router always functions as the authenticator. When you configure PPP authentication in a dynamic profile, the **chap** and **pap** statements do not support any additional configuration options.

   - To configure CHAP authentication:

   ```
   [edit dynamic-profiles basic-pppoe-profile interfaces pp0 unit "$junos-interface-unit"]
   user@host# set ppp-options chap
   ```

   - To configure PAP authentication:

   ```
   [edit dynamic-profiles basic-pppoe-profile interfaces pp0 unit "$junos-interface-unit"]
   user@host# set ppp-options pap
   ```

6. Configure the protocol family for the **pp0** interface.

   a. Specify that you want to configure the **inet** (IPv4) or **inet6** (IPv6) protocol family.

   ```
   [edit dynamic-profiles basic-pppoe-profile interfaces pp0 unit
     "$junos-interface-unit"]
   user@host# edit family inet
   ```

   b. Configure the unnumbered address for the protocol family.

   ```
   [edit dynamic-profiles basic-pppoe-profile interfaces pp0 unit "$junos-interface-unit"
     family inet]
   user@host# set unnumbered-address lo0.0
   ```

Related Documentation

- Subscriber Interfaces and PPPoE Overview on page 141
- Dynamic PPPoE Subscriber Interfaces over Static Underlying Interfaces Overview on page 153
- Configuring an Underlying Interface for Dynamic PPPoE Subscriber Interfaces on page 156
- Example: Configuring a Dynamic PPPoE Subscriber Interface on a Static Gigabit Ethernet VLAN Interface on page 162
- Verifying and Managing Dynamic PPPoE Configuration on page 211

Configuring a PPPoE Dynamic Profile with Additional Options

You can configure a dynamic profile for PPPoE that has additional options for subscriber access. All of these optional statements, with the exception of the **keepalives** and **no-keepalives** statements, are configured at the **[edit dynamic-profiles *profile-name* interfaces pp0 unit "$junos-interface-unit" family *family*]** hierarchy level. The **keepalives** and **no-keepalives** statements are configured at the **[edit dynamic-profiles *profile-name* interfaces pp0 unit "$junos-interface-unit"]** hierarchy level.

The additional options for PPPoE subscriber access in a dynamic profile can include one or more of the following:

- The keepalive interval (**keepalives**), or the option to disable sending keepalive messages (**no-keepalives**)
- The IPv4 or IPv6 address of the dynamic PPPoE logical interface (**address**)
- Definition of the service sets and filters to be applied to the dynamic PPPoE logical interface, configured at the **[edit dynamic-profiles *profile-name* interfaces pp0 unit "$junos-interface-unit" family *family* service]** hierarchy level
- Association of an input and output filter to the dynamic PPPoE logical interface, configured at the **[edit dynamic-profiles *profile-name* interfaces pp0 unit "$junos-interface-unit" family *family* filter]** hierarchy level

NOTE: The creation of dynamic PPPoE subscriber interfaces is supported for the inet and inet6 protocol families.

Before you begin:

- Configure a basic PPPoE dynamic profile.

  See "Configuring a Basic PPPoE Dynamic Profile" on page 146.

To configure a PPPoE dynamic profile with additional options for subscriber access:

1. Modify the keepalive interval, or configure the router to disable sending keepalive messages.
   - To modify the keepalive interval:

     ```
     [edit dynamic-profiles business-pppoe-profile interfaces pp0 unit
       "$junos-interface-unit"]
     user@host# set keepalives interval 15
     ```
   - To disable sending keepalive messages:

     ```
     [edit dynamic-profiles business-pppoe-profile interfaces pp0 unit
       "$junos-interface-unit"]
     user@host# set no-keepalives
     ```
2. Specify that you want to configure the **inet** (IPv4) or **inet6** (IPv6) protocol family.


```
[edit dynamic-profiles business-pppoe-profile interfaces pp0 unit
  "$junos-interface-unit"]
user@host# edit family inet
```

3. Specify the IPv4 or IPv6 address of the dynamic PPPoE logical interface.

```
[edit dynamic-profiles business-pppoe-profile interfaces pp0 unit
  "$junos-interface-unit" family inet]
user@host# set address 6.6.6.7/32
```

4. Specify the input and output service sets that you want to apply to the dynamic PPPoE logical interface.

```
[edit dynamic-profiles business-pppoe-profile interfaces pp0 unit
  "$junos-interface-unit" family inet]
user@host# set service input service-set inputService_100
user@host# set service input post-service-filter postService_20
user@host# set service output service-set outputService_200
```

5. Specify the input and output filters that you want to apply to the dynamic PPPoE logical interface.

   To control the order in which filters are processed, you can optionally specify a precedence value for the input filter, output filter, or both.

```
[edit dynamic-profiles business-pppoe-profile interfaces pp0 unit
  "$junos-interface-unit" family inet]
user@host# set filter input pppoe-input-filter
user@host# set filter output pppoe-output-filter precedence 50
```

Related Documentation

- Subscriber Interfaces and PPPoE Overview on page 141
- Dynamic PPPoE Subscriber Interfaces over Static Underlying Interfaces Overview on page 153
- Configuring a Basic PPPoE Dynamic Profile on page 146
- Configuring an Underlying Interface for Dynamic PPPoE Subscriber Interfaces on page 156
- Example: Configuring a Dynamic PPPoE Subscriber Interface on a Static Gigabit Ethernet VLAN Interface on page 162
- Verifying and Managing Dynamic PPPoE Configuration on page 211

- *Dynamic Service Sets Overview*
- *Associating Service Sets with Interfaces in a Dynamic Profile*
- *Understanding Dynamic Firewall Filters*
- *Dynamically Attaching Statically Created Filters for a Specific Interface Family Type*

Assigning a Dynamic Profile and Routing Instance to a Service Name or ACI/ARI Pair for Dynamic PPPoE Interface Creation

You can create a dynamic PPPoE subscriber interface based on the service name, agent circuit identifier (ACI), and agent remote identifier (ARI) information provided by the PPPoE client during PPPoE negotiation. To do so, you assign a previously configured PPPoE dynamic profile to a named service, **empty** service, or **any** service entry in a PPPoE service name table, or to an ACI/ARI pair defined for these services.

Similarly, to specify the routing instance in which to instantiate the dynamic PPPoE subscriber interface, you can assign a previously configured routing instance to a named service, **empty** service, or **any** service in a PPPoE service name table, or to an ACI/ARI pair defined for these services.

Observe the following configuration guidelines when you assign a dynamic profile and routing instance to a PPPoE service name table to create a dynamic PPPoE subscriber interface:

- The dynamic profile and routing instance must already be configured on the router.
- The dynamic profile or routing instance assigned to the PPPoE service name table overrides the dynamic profile or routing instance assigned to the PPPoE underlying interface on which the dynamic subscriber interface is created.
- You cannot configure a dynamic profile or routing instance for an ACI/ARI pair already configured with a static interface (by using the **static-interface** statement). Conversely, you cannot configure a static interface for an ACI/ARI pair already configured with a dynamic profile or routing instance.

Before you begin:

1. Configure a PPPoE dynamic profile in either of the following ways:
   - To configure a basic PPPoE dynamic profile, see "Configuring a Basic PPPoE Dynamic Profile" on page 146.
   - To configure a PPPoE dynamic profile with additional options for subscriber access, see "Configuring a PPPoE Dynamic Profile with Additional Options" on page 148.
2. Configure the routing instance in which you want the router to instantiate the dynamic profile.

   For information about configuring routing instances, see *Routing Instances Overview*.
3. Create the PPPoE service name table on the router.

   See *Creating a Service Name Table*.

To create a dynamic PPPoE subscriber interface based on the service name and, optionally, associated ACI/ARI pair configured in a PPPoE service name table, do one of the following:

- Assign a previously configured dynamic profile and routing instance to a named, **empty**, or **any** service.


```
[edit protocols pppoe service-name-tables table1]
user@host# set service premium dynamic-profile premiumProfile routing-instance
  premiumRI
```

- Assign a previously configured dynamic profile and routing instance to the ACI/ARI pair defined for a named, **empty**, or **any** service.

```
[edit protocols pppoe service-name-tables table1]
user@host# set service any agent-specifier aci west-ge-3/0/3 ari sunnyvale
  dynamic-profile standardProfile routing-instance standardRI
```

Related Documentation

- Example: Configuring a PPPoE Service Name Table for Dynamic Subscriber Interface Creation on page 207
- Subscriber Interfaces and PPPoE Overview on page 141
- Configuring Dynamic PPPoE Subscriber Interfaces Using Dynamic Profiles on page 145
- *Configuring PPPoE Service Name Tables*

CHAPTER 16

Configuring PPPoE Subscriber Interfaces and Underlying Interfaces

Dynamic PPPoE Subscriber Interfaces over Static Underlying Interfaces Overview

Creating a dynamic PPPoE subscriber interface over a static underlying Ethernet interface consists of two basic steps:

1. Configure a dynamic profile to define the attributes of the PPPoE logical interface.
2. Attach the dynamic profile to a statically created underlying Ethernet interface configured with PPPoE encapsulation.

This overview describes the concepts you need to understand to configure a dynamic PPPoE subscriber interface, and covers the following topics:

PPPoE Dynamic Profile Configuration

A *dynamic profile* is a template for configuring a dynamic interface. You use predefined dynamic variables in the PPPoE dynamic profile to represent information that varies from subscriber to subscriber, such as the logical unit number and underlying interface name. These variables are dynamically replaced with the values supplied by the network when the subscriber logs in. On receipt of traffic on an underlying Ethernet interface to which a dynamic profile is attached, the router creates the dynamic PPPoE logical interface, also referred to as a *dynamic PPPoE subscriber interface*, on the underlying interface and applies the properties configured in the dynamic profile.

To provide basic access for PPPoE subscribers, the dynamic profile must provide a minimal configuration for a **pp0** (PPPoE) logical interface that includes at least the following attributes:

- The logical unit number, represented by the **$junos-interface-unit** predefined dynamic variable
- The name of the underlying Ethernet interface, represented by the **$junos-underlying-interface** predefined dynamic variable
- Configuration of the router to act as a PPPoE server
- The PPP authentication protocol (PAP or CHAP)
- The unnumbered address for the **inet** (IPv4) or **inet6** (IPv6) protocol family

You can also optionally configure additional options for PPPoE subscriber access in the dynamic profile, including:

- The keepalive interval, or the option to disable sending keepalive messages
- The IPv4 or IPv6 address of the dynamic PPPoE logical interface
- The service sets and filters, input filters, and output filters to be applied to the dynamic PPPoE logical interface

PPPoE Underlying Interface Configuration

After you configure a dynamic profile to define the attributes of a dynamic PPPoE subscriber interface, you must attach the dynamic profile to the underlying Ethernet interface on which you want the router to dynamically create the PPPoE logical interface. The underlying interface for a dynamic PPPoE logical interface must be statically created and configured with PPPoE (**ppp-over-ether**) encapsulation. When a PPPoE subscriber logs in on the underlying interface, the router dynamically creates the PPPoE logical interface and applies the attributes defined in the profile to the interface.

In addition to attaching the dynamic profile to the interface, you can also configure the underlying interface with one or more of the following optional PPPoE-specific attributes:

- Prevention of another dynamic PPPoE logical interface from being activated on the underlying interface when a PPPoE logical interface for a client with the same MAC address is already active on that interface
- Maximum number of dynamic PPPoE logical interfaces (sessions) that the router can activate on the underlying interface
- An alternative access concentrator name in the AC-NAME tag in a PPPoE control packet

Address Assignment for Dynamic PPPoE Subscriber Interfaces

If the subscriber address for a dynamic PPPoE interface is not specified by means of the Framed-IP-Address (8) or Framed-Pool (88) RADIUS IETF attributes during authentication, the router allocates an IP address from the first IPv4 local address-assignment pool defined in the routing instance. For this reason, make sure that the local address assigned for the **inet** (IPv4) address family is in the same subnet as the addresses obtained from the first IPv4 local address-assignment pool.

The router allocates the IP address from the first IPv4 local address-assignment pool under either of the following conditions:

- RADIUS returns no address attributes.
- RADIUS authentication does not take place because only address allocation is requested.

If the first IPv4 local address-assignment pool has no available addresses, or if no IPv4 local address-assignment pools are configured, the router does not allocate an IP address to the dynamic PPPoE subscriber interface, and denies access to the associated subscriber. To avoid depletion of IP addresses, you can configure linked address-assignment pools on the first IPv4 local address-assignment pool to create one or more backup pools.

For more information, see *Configuring Address-Assignment Pools*.

Guidelines for Configuring Dynamic PPPoE Subscriber Interfaces

Observe the following guidelines when you configure dynamic PPPoE subscriber interfaces:

- You can configure dynamic PPPoE subscriber interfaces for the **inet** (IPv4) and **inet6** (IPv6) protocol families.
- When you configure the **pp0** (PPPoE) logical interface in a PPPoE dynamic profile, you must include the **pppoe-options** subhierarchy at the **[edit dynamic-profiles *profile-name* interfaces pp0 unit "$junos-interface-unit"]** hierarchy level. At a minimum, the **pppoe-options** subhierarchy must include the name of the underlying Ethernet interface, represented by the **$junos-underlying-interface** predefined dynamic variable, and the **server** statement, which configures the router to act as a PPPoE server. If you omit the **pppoe-options** subhierarchy from the configuration, the **commit** operation fails.
- When you configure CHAP or PAP authentication in a PPPoE dynamic profile, you cannot configure additional options for the **chap** or **pap** statements. This is because the router supports only unidirectional authentication for dynamic interfaces; that is, the router always functions as the authenticator.
- When you attach the PPPoE dynamic profile to an underlying Ethernet interface, ensure that both of the following conditions are met:
  - The PPPoE dynamic profile has already been configured on the router.

  - The underlying Ethernet interface has already been statically configured on the router with PPPoE (**ppp-over-ether**) encapsulation.

- You cannot attach a PPPoE dynamic profile to an underlying Ethernet interface that is already associated with static PPPoE logical interfaces. Conversely, you cannot associate static PPPoE logical interfaces with an underlying Ethernet interface that already has a PPPoE dynamic profile attached.

Related Documentation

- Subscriber Interfaces and PPPoE Overview on page 141
- Configuring Dynamic PPPoE Subscriber Interfaces Using Dynamic Profiles on page 145
- Example: Configuring a Dynamic PPPoE Subscriber Interface on a Static Gigabit Ethernet VLAN Interface on page 162
- Understanding PPPoE Service Name Tables on page 201

Configuring an Underlying Interface for Dynamic PPPoE Subscriber Interfaces

After you configure a dynamic profile to define the attributes of a dynamic PPPoE subscriber interface, you must attach the dynamic profile to a statically created underlying Ethernet interface configured with PPPoE (**ppp-over-ether**) encapsulation. You configure the underlying interface at the **[edit interfaces *interface-name* unit *logical-unit-number* pppoe-underlying-options]** hierarchy level.

In addition to attaching the dynamic profile to the interface by using the required **dynamic-profile** statement, you can also configure the underlying interface with one or more of the following optional PPPoE-specific attributes:

- Prevention of another dynamic PPPoE logical interface from being activated on the underlying interface when a PPPoE logical interface for a client with the same MAC address is already active on that interface (**duplicate-protection**)
- Maximum number of dynamic PPPoE logical interfaces (sessions) that the router can activate on the underlying interface on a per-interface basis (using the **max-sessions** statement) or on a per-subscriber basis (using the Max-Clients-Per-Interface Juniper Networks VSA [26-143])
- Lockout of failed or short-lived (also known as short-cycle) PPPoE subscriber sessions to prevent reconnection to the router for a default or configurable period of time (**short-cycle-protection**)
- An alternative access concentrator name in the AC-NAME tag in a PPPoE control packet (**access-concentrator**)
- Ability to ignore DSL Forum VSAs received in PPPoE control packets when the router is directly connected to a CPE device (**direct-connect**).

Before you begin:

1. Configure the static underlying Ethernet interface on which you want the router to dynamically create the PPPoE logical interface.

For information about configuring static Ethernet interfaces, see *Configuring Ethernet Physical Interface Properties*.

2. Configure a PPPoE dynamic profile in either of the following ways:
   - To configure a basic PPPoE dynamic profile, see "Configuring a Basic PPPoE Dynamic Profile" on page 146.
   - To configure a PPPoE dynamic profile with additional options for subscriber access, see "Configuring a PPPoE Dynamic Profile with Additional Options" on page 148.

To configure an underlying Ethernet interface for a dynamic PPPoE subscriber interface:

1. Specify the name and logical unit number of the static underlying Ethernet interface to which you want to attach the PPPoE dynamic profile.

   ```
   [edit]
   user@host# edit interfaces ge-1/0/1 unit 0
   ```
2. Configure PPPoE encapsulation on the underlying interface.

   ```
   [edit interfaces ge-1/0/1 unit 0]
   user@host# set encapsulation ppp-over-ether
   ```
3. Specify that you want to configure PPPoE-specific options on the underlying interface.

   ```
   [edit interfaces ge-1/0/1 unit 0]
   user@host# edit pppoe-underlying-options
   ```
4. Attach a previously configured PPPoE dynamic profile to the underlying interface.

   The specified PPPoE dynamic profile must already be configured on the router. In addition, you cannot attach a PPPoE dynamic profile to an underlying Ethernet interface that is already associated with static PPPoE logical interfaces. Conversely, you cannot associate static PPPoE logical interfaces with an underlying Ethernet interface that already has a PPPoE dynamic profile attached.

   ```
   [edit interfaces ge-1/0/1 unit 0 pppoe-underlying-options]
   user@host# set dynamic-profile basic-pppoe-profile
   ```
5. (Optional) Enable duplicate protection to prevent activation of another dynamic PPPoE logical interface for the same client on the underlying interface.

   ```
   [edit interfaces ge-1/0/1 unit 0 pppoe-underlying-options]
   user@host# set duplicate-protection
   ```
6. (Optional) Configure the maximum number of concurrent PPPoE sessions that the router can activate on the underlying interface in either of the following ways:
   - To configure the maximum number of concurrent PPPoE sessions on a per-interface basis, from 1 to the platform-specific default for your router, use the **max-sessions** statement:

     ```
     [edit interfaces ge-1/0/1 unit 0 pppoe-underlying-options]
     user@host# set max-sessions 20
     ```
   - To configure the maximum number of concurrent PPPoE sessions on a per-subscriber basis, use the value returned by RADIUS in the Max-Clients-Per-Interface Juniper Networks vendor-specific attribute (VSA)

[26-143]. By default, the PPPoE maximum session value returned by RADIUS in the Max-Clients-Per-Interface VSA takes precedence over the PPPoE maximum session value configured with the **max-sessions** statement.

7. (Optional) Configure the router to ignore the value returned by RADIUS in the Max-Clients-Per-Interface VSA and restore the PPPoE maximum session value on the underlying interface to the value configured in the CLI with the **max-sessions** statement.

   ```
   [edit interfaces ge-1/0/1 unit 0 pppoe-underlying-options]
   user@host# set max-sessions-vsa-ignore
   ```

8. (Optional) Enable PPPoE subscriber session lockout on the PPPoE underlying interface in either of the following ways:

   - To configure PPPoE subscriber session lockout with the default lockout period:

     ```
     [edit interfaces interface-name unit logical-unit-number pppoe-underlying-options]
     user@host# set short-cycle-protection
     ```

   - To configure PPPoE subscriber session lockout with a nondefault lockout period:

     ```
     [edit interfaces interface-name unit logical-unit-number pppoe-underlying-options]
     user@host# set short-cycle-protection lockout-time-min minimum-seconds
       lockout-time-max maximum-seconds
     ```

   BEST PRACTICE: When you configure PPPoE subscriber session lockout, we recommend that you also enable duplicate protection to ensure that the MAC source address for each PPPoE session is unique on the underlying interface.

9. (Optional) Specify the alternative name for the access concentrator, also known as the PPPoE server.

   ```
   [edit interfaces ge-1/0/1 unit 0 pppoe-underlying-options]
   user@host# set access-concentrator server-east
   ```

10. (Optional) Specify that DSL Forum VSAs received on the interface are ignored; use when the CPE device is directly connected to the router.

    ```
    [edit interfaces ge-1/0/1 unit 0 pppoe-underlying-options]
    user@host# set direct-connect
    ```

Related Documentation

- Subscriber Interfaces and PPPoE Overview on page 141
- Dynamic PPPoE Subscriber Interfaces over Static Underlying Interfaces Overview on page 153
- Configuring Dynamic PPPoE Subscriber Interfaces Using Dynamic Profiles on page 145
- Configuring the PPPoE Family for an Underlying Interface on page 159
- Configuring Lockout of PPPoE Subscriber Sessions on page 198
- Verifying and Managing Dynamic PPPoE Configuration on page 211

- Example: Configuring a Dynamic PPPoE Subscriber Interface on a Static Gigabit Ethernet VLAN Interface on page 162
- *Configuring Ethernet Physical Interface Properties*

Configuring the PPPoE Family for an Underlying Interface

You can configure the PPPoE family on an underlying interface as an alternative to configuring PPPoE encapsulation on that interface. You cannot configure both on the same interface. You can configure the same attributes for the PPPoE family as you can for an interface configured with **pppoe-underlying-options**.

Before you begin, configure the underlying interface. When you want to configure PPPoE on an aggregated Ethernet bundle, you must configure the PPPoE family over a VLAN demux interface as an intermediate underlying option. The VLAN demux interface can be static or dynamic.

The following topics describe how to configure basic static and dynamic interfaces:

- *Configuring a Subscriber Interface with a Static VLAN Interface*
- *Configuring Static Subscriber Interfaces Using VLAN Demux Interfaces*
- Configuring Dynamic Subscriber Interfaces Using VLAN Demux Interfaces in Dynamic Profiles on page 79

To configure the PPPoE family over an underlying interface:

1. Specify the PPPoE family.

   [edit interfaces demux0 unit *logical-unit-number*]
   user@host# **set family pppoe**

2. (Optional) Configure an alternative access concentrator name to be used instead of the system name in PPPoE control packets for the dynamic PPPoE subscriber interface.

   [edit interfaces demux0 unit *logical-unit-number* family pppoe]
   user@host# **set access-concentrator *name***

3. (Optional) Configure duplicate protection to prevent the activation of another dynamic PPPoE logical interface on the same underlying interface when a dynamic PPPoE logical interface for a client with the same MAC address is already active on that interface.

   [edit interfaces demux0 unit *logical-unit-number* family pppoe]
   user@host# **set duplicate-protection**

4. (Optional) Attach a dynamic profile to determine the properties of the dynamic PPPoE logical interface when it is created.

   [edit interfaces demux0 unit *logical-unit-number* family pppoe]
   user@host# **set dynamic-profile *profile-name***

5. (Optional) Configure the maximum number of concurrent PPPoE sessions that the router can activate on the underlying interface in either of the following ways:

- To configure the maximum number of concurrent PPPoE sessions on a per-interface basis, from 1 through the platform-specific default for your router, use the **max-sessions** statement:

  [edit interfaces demux0 unit *logical-unit-number* family pppoe]
  user@host# **set** max-sessions ***number***

- To configure the maximum number of concurrent PPPoE sessions on a per-subscriber basis, use the value returned by RADIUS in the Max-Clients-Per-Interface Juniper Networks vendor-specific attribute (VSA) [26-143]. By default, the PPPoE maximum session value returned by RADIUS in the Max-Clients-Per-Interface VSA takes precedence over the PPPoE maximum session value configured with the **max-sessions** statement.

6. (Optional) Configure the router to ignore the value returned by RADIUS in the Max-Clients-Per-Interface VSA and restore the PPPoE maximum session value on the underlying interface to the value configured in the CLI with the **max-sessions** statement.

   [edit interfaces demux0 unit *logical-unit-number* family pppoe]
   user@host# **set** max-sessions-vsa-ignore

7. (Optional) Enable PPPoE subscriber session lockout on the PPPoE underlying interface in either of the following ways:

   - To configure PPPoE subscriber session lockout with the default lockout period:

     [edit interfaces *interface-name* unit *logical-unit-number* pppoe-underlying-options]
     user@host# **set** short-cycle-protection

   - To configure PPPoE subscriber session lockout with a nondefault lockout period:

     [edit interfaces *interface-name* unit *logical-unit-number* pppoe-underlying-options]
     user@host# **set** short-cycle-protection **lockout-time-min *minimum-seconds***
     lockout-time-max *maximum-seconds*

   BEST PRACTICE: When you configure PPPoE subscriber session lockout, we recommend that you also enable duplicate protection to ensure that the MAC source address for each PPPoE session is unique on the underlying interface.

8. (Optional) Specify the service name table assigned to the underlying interface.

   [edit interfaces demux0 unit *logical-unit-number* family pppoe]
   user@host# **set** service-name-table ***table-name***

9. (Optional) Specify that DSL Forum VSAs received on the interface are ignored; use when the CPE device is directly connected to the router.

   [edit interfaces demux0 unit *logical-unit-number* family pppoe]
   user@host# **set** direct-connect

Related Documentation

- Static or Dynamic Demux Subscriber Interfaces over Aggregated Ethernet Overview on page 89

- Configuring an Underlying Interface for Dynamic PPPoE Subscriber Interfaces on page 156
- Configuring Lockout of PPPoE Subscriber Sessions on page 198
- Example: Configuring a Static PPPoE Subscriber Interface on a Static Underlying VLAN Demux Interface over Aggregated Ethernet on page 165
- Example: Configuring a Dynamic PPPoE Subscriber Interface on a Static Underlying VLAN Demux Interface over Aggregated Ethernet on page 171
- Example: Configuring a Dynamic PPPoE Subscriber Interface on a Dynamic Underlying VLAN Demux Interface over Aggregated Ethernet on page 176

Ignoring DSL Forum VSAs from Directly Connected Devices

When CPE devices are directly connected to a BNG, you may want the router to ignore any DSL Forum VSAs that it receives in PPPoE control packets because the VSAs can be spoofed by malicious subscribers. Spoofing is particularly serious when the targeted VSAs are used to authenticate the subscriber, such as Agent-Circuit-Id [26-1] and Agent-Remote-ID [26-2]. You can include the **direct-connect** statement to ignore DSL Forum VSAs on static or dynamic PPPoE interfaces or PPPoE underlying interfaces.

To configure the router to ignore DSL Forum VSAs on specific PPPoE interfaces:

1. Specify that you want to configure PPPoE-specific options on the interface:
   - For a PPPoE family in a dynamic profile for a VLAN demultiplexing (demux) logical interface:

     [edit dynamic-profiles *profile-name* interfaces demux0 unit *logical-unit-number*]
     user@host# **edit family pppoe**

   - For a PPPoE family in a dynamic profile:

     [edit dynamic-profiles *profile-name* interfaces *interface-name* unit *logical-unit-number*]
     user@host# **edit family pppoe**

   - For a PPPoE underlying interface in a dynamic profile:

     [edit dynamic-profiles *profile-name* interfaces *interface-name* unit *logical-unit-number*]
     user@host# **edit pppoe-underlying-options**

   - For a PPPoE family on an underlying interface:

     [edit interfaces *interface-name* unit *logical-unit-number*]
     user@host# **edit family pppoe**

   - For an underlying interface with PPPoE encapsulation:

     [edit interfaces *interface-name* unit *logical-unit-number*]
     user@host# **edit pppoe-underlying-options**

2. Specify that the router ignores DSL forum VSAs received on a specific interface.

   [edit ... family pppoe]
   user@host# **set direct-connect**

   or

```
[edit ... pppoe-underlying-options]
user@host# set direct-connect
```

Related Documentation

- Configuring an Underlying Interface for Dynamic PPPoE Subscriber Interfaces on page 156
- Configuring the PPPoE Family for an Underlying Interface on page 159

Example: Configuring a Dynamic PPPoE Subscriber Interface on a Static Gigabit Ethernet VLAN Interface

This example shows how to configure a dynamic PPPoE subscriber interface on a statically configured Gigabit Ethernet VLAN underlying interface. When a PPPoE subscriber logs in on the underlying interface, the router creates the dynamic PPPoE subscriber interface with the attributes specified in the dynamic profile.

In this example, the dynamic PPPoE profile, **pppoe-profile-east**, defines options for PPPoE subscribers accessing the network, and includes the predefined dynamic variables **$junos-interface-unit**, which represents the logical unit number of the dynamic PPPoE logical interface, and **$junos-underlying-interface**, which represents the name of the underlying Ethernet interface. The **pppoe-profile-east** dynamic profile is assigned to the underlying Ethernet VLAN interface **ge-2/0/3.1** that is configured with PPPoE (**ppp-over-ether**) encapsulation.

When the router dynamically creates the PPPoE subscriber interface on **ge-2/0/3.1** in response to a subscriber login, the values of **$junos-interface-unit** and **$junos-underlying-interface** are dynamically replaced with the actual logical unit number and interface name, respectively, that are supplied by the network when the PPPoE subscriber logs in.

To configure a dynamic PPPoE subscriber interface:

1. Configure a dynamic profile to define the attributes of the dynamic PPPoE subscriber interface.

```
[edit]
dynamic-profiles {
  pppoe-profile-east {
    interfaces {
      pp0 {
        unit "$junos-interface-unit" {
          ppp-options {
            chap;
          }
          pppoe-options {
            underlying-interface "$junos-underlying-interface";
            server;
          }
          keepalives interval 30;
          family inet {
            filter {
              input pppoe-input-filter-east;
```

```
                    output pppoe-output-filter-east precedence 20;
                  }
                  service {
                    input {
                      service-set inputService-east;
                      post-service-filter postService-east;
                    }
                    output {
                      service-set outputService-east;
                    }
                  }
                  address 6.6.6.1/32;
                  unnumbered-address lo0.0;
                }
              }
            }
          }
        }
      }
```

2. Assign the dynamic PPPoE profile to the static underlying Ethernet interface, and define PPPoE-specific attributes for the underlying interface.

```
[edit]
interfaces {
  ge-2/0/3 {
    vlan-tagging;
    unit 1 {
      encapsulation ppp-over-ether;
      vlan-id 100;
      pppoe-underlying-options {
        access-concentrator server-east;
        duplicate-protection;
        dynamic-profile pppoe-profile-east;
        max-sessions 10;
      }
    }
  }
}
```

Related Documentation

- Subscriber Interfaces and PPPoE Overview on page 141
- Dynamic PPPoE Subscriber Interfaces over Static Underlying Interfaces Overview on page 153
- Configuring a PPPoE Dynamic Profile with Additional Options on page 148
- Configuring an Underlying Interface for Dynamic PPPoE Subscriber Interfaces on page 156

CHAPTER 17

Configuring PPPoE Subscriber Interfaces over Aggregated Ethernet Examples

Example: Configuring a Static PPPoE Subscriber Interface on a Static Underlying VLAN Demux Interface over Aggregated Ethernet

This example shows how you can configure static PPPoE subscriber interfaces over aggregated Ethernet bundles to provide subscriber link redundancy.

Requirements

PPPoE over VLAN demux interfaces over aggregated Ethernet requires the following hardware and software:

- MX Series 3D Universal Edge Routers
- MPCs
- Junos OS Release 11.2 or later

No special configuration beyond device initialization is required before you can configure this feature.

Overview

Aggregated Ethernet bundles enable link redundancy between the router and networking devices connected by Ethernet links. This example describes how to configure link redundancy for static PPPoE subscribers over aggregated Ethernet interface with an intermediate static VLAN demux interface. Sample tasks include configuring a two-member aggregated Ethernet bundle on **ae0**, configuring a static VLAN demux interface, **demux0.100**, that underlies the PPPoE subscriber interface, **pp0.100**, and configuring the PPPoE subscriber interface including characteristics of the PPPoE family.

This example does not show all possible configuration choices.

Configuration

CLI Quick Configuration

To quickly configure link redundancy for static PPPoE subscribers over a static VLAN demux interface over aggregated Ethernet, copy the following commands, paste them in a text file, remove any line breaks, and then copy and paste the commands into the CLI.

```
[edit]
set chassis aggregated-devices ethernet device-count 1
set interfaces ge-5/0/3 gigether-options 802.3ad ae0
set interfaces ge-5/0/3 gigether-options 802.3ad primary
set interfaces ge-5/1/2 gigether-options 802.3ad ae0
set interfaces ge-5/1/2 gigether-options 802.3ad backup
set interfaces ae0 flexible-vlan-tagging
set interfaces ae0 aggregated-ether-options link-protection
edit interfaces demux0 unit 100
set vlan-id 100
set demux-options underlying-interface ae0
set family pppoe access-concentrator pppoe-server-1
set family pppoe duplicate-protection
set family pppoe max-sessions 16000
top
edit interfaces pp0 unit 100
set pppoe-options underlying-interface demux0.100
set pppoe-options server
set family inet unnumbered-address lo0.0
top
```

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 link redundancy for static PPPoE subscribers over a static VLAN demux interface over aggregated Ethernet:

1. Define the number of aggregated Ethernet devices on the router.

   ```
   [edit chassis]
   user@host# set aggregated-devices ethernet device-count 1
   ```

2. Configure a two-link aggregated Ethernet logical interface to serve as the underlying interface for the static VLAN demux subscriber interface. In this example, the LAG bundle is configured for one-to-one active/backup link redundancy. To support link redundancy at the MPC level, the LAG bundle attaches to ports from two different MPCs.

   ```
   [edit interfaces]
   ```

```
user@host# set ge-5/0/3 gigether-options 802.3ad ae0
user@host# set ge-5/0/3 gigether-options 802.3ad primary
user@host# set ge-5/1/2 gigether-options 802.3ad ae0
user@host# set ge-5/1/2 gigether-options 802.3ad backup
```

3. Enable link protection on the aggregated Ethernet logical interface and configure support for single and dual (stacked) VLAN tags.

```
[edit interfaces]
user@host# set ae0 aggregated-ether-options link-protection
user@host# set ae0 flexible-vlan-tagging
```

4. Configure the VLAN demux interface over the aggregated Ethernet logical interface.

```
[edit interfaces]
user@host# set demux0 unit 100 vlan-id 100
user@host# set demux0 unit 100 demux-options underlying-interface ae0
```

5. Configure the PPPoE family attributes on the VLAN demux interface.

```
[edit interfaces]
user@host# set demxu0 unit 100 family pppoe access-concentrator pppoe-server-1
user@host# set demux0 unit 100 family pppoe duplicate-protection
user@host# set demux0 unit 100 family pppoe max-sessions 16000
```

6. Configure the VLAN demux interface as the underlying interface on which the PPPoE logical interface is created.

```
[edit interfaces]
user@host# set pp0 unit 100 pppoe-options underlying-interface demux0.100
user@host# set pp0 unit 100 pppoe-options server
user@host# set pp0 unit 100 family inet unnumbered-address lo0.0
```

Results From configuration mode, confirm the aggregated device configuration by entering the **show chassis** command. Confirm the interface 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 chassis
aggregated-devices {
  ethernet {
    device-count 1;
  }
}
```

```
[edit]
user@host# show interfaces
  ge-5/0/3 {
    gigether-options {
      802.3ad {
        ae0;
        primary;
      }
    }
  }
  ge-5/1/2 {
    gigether-options {
```

```
            802.3ad {
                ae0;
                backup;
            }
        }
    }
    ae0 {
        flexible-vlan-tagging;
        aggregated-ether-options {
            link-protection;
        }
    }
    demux0 {
        unit 100 {
            vlan-id 100;
            demux-options {
                underlying-interface ae0;
            }
            family pppoe {
                access-concentrator pppoe-server-1;
                duplicate-protection;
                max-sessions 16000;
            }
        }
    }
    pp0 {
        unit 100 {
            pppoe-options {
                underlying-interface demux0.100;
                server;
            }
            family inet {
                unnumbered-address lo0.0;
            }
        }
    }
```

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

Verification

To confirm that the configuration is working properly, perform these tasks:

- Verifying the Aggregated Ethernet Interface Configuration on page 168
- Verifying the demux0 Interface Configuration on page 169
- Verifying the pp0 Interface Configuration on page 170

Verifying the Aggregated Ethernet Interface Configuration

Purpose Verify that the interface values match your configuration, the link is up, and traffic is flowing.

Action From operational mode, enter the **show interfaces redundancy** command.

```
user@host> show interfaces redundancy
Interface  State            Last change  Primary      Secondary    Current status
ae0        On primary                     ge-5/0/3     ge-5/1/2     both up
```

From operational mode, enter the **show interfaces ae0** command.

```
user@host> show interfaces ae0
Physical interface: ae0, Enabled, Physical link is Up
  Interface index: 128, SNMP ifIndex: 606
  Link-level type: Ethernet, MTU: 1522, Speed: 1Gbps, BPDU Error: None,
  MAC-REWRITE Error: None, Loopback: Disabled, Source filtering: Disabled,
  Flow control: Disabled, Minimum links needed: 1, Minimum bandwidth needed: 0
  Device flags   : Present Running
  Interface flags: SNMP-Traps Internal: 0x4000
  Current address: 00:1f:12:b8:ef:c0, Hardware address: 00:1f:12:b8:ef:c0
  Last flapped   : 2011-03-11 13:24:18 PST (2d 03:34 ago)
  Input rate     : 1984 bps (2 pps)
  Output rate    : 0 bps (0 pps)

  Logical interface ae0.32767 (Index 69) (SNMP ifIndex 709)
    Flags: SNMP-Traps 0x4004000 VLAN-Tag [ 0x0000.0 ]  Encapsulation: ENET2
    Statistics        Packets        pps         Bytes          bps
    Bundle:
        Input :        371259          2      46036116         1984
        Output:             0          0             0            0
    Protocol multiservice, MTU: Unlimited
      Flags: Is-Primary
```

Meaning The **show interfaces redundancy** output shows the redundant link configuration and that both link interfaces are up. The **show interfaces ae0** output shows that the aggregated Ethernet interface is up and that traffic is being received on the logical interface.

Verifying the demux0 Interface Configuration

Purpose Verify that the VLAN demux interface displays the configured PPPoE family attributes and the member links in the aggregated Ethernet bundle.

Action From operational mode, enter the **show interfaces demux0** command.

```
user@host> show interfaces demux0.100
Logical interface demux0.100 (Index 76) (SNMP ifIndex 61160)
    Flags: SNMP-Traps 0x4000 VLAN-Tag [ 0x8100.100 ]
    Encapsulation: ENET2
    Demux:
      Underlying interface: ae0 (Index 199)
    Link:
      ge-5/0/3
      ge-5/1/2
    Input packets : 2
    Output packets: 18575
    Protocol pppoe
      Dynamic Profile: none,
      Service Name Table: None,
      Max Sessions: 16000, Duplicate Protection: On,
      AC Name: pppoe-server-1
```

Alternatively, you can enter **show pppoe underlying-interfaces detail** to display the state and PPPoE family configuration for all configured underlying interfaces.

Meaning The output shows the name of the underlying interface, the member links of the aggregated bundle, and the PPPoE family configuration. The output shows packet counts when traffic is present on the logical interface.

Verifying the pp0 Interface Configuration

Purpose Verify that the interface values match your configuration.

Action From operational mode, enter the **show interfaces pp0** command.

```
user@host> show interfaces pp0.100
  Logical interface pp0.100 (Index 71) (SNMP ifIndex 710)
    Flags: Point-To-Point SNMP-Traps 0x4000 Encapsulation: PPPoE
    PPPoE:
      State: SessionUp, Session ID: 1,
      Session AC name: pppoe-server-1, Remote MAC address: 00:90:1a:00:18:34,
      Underlying interface: demux0.100 (Index 70)
    Link:
      ge-5/0/3.32767
      ge-5/1/2.32767
    Input packets : 18572
    Output packets: 18572
  Keepalive settings: Interval 10 seconds, Up-count 1, Down-count 3
  Keepalive: Input: 0 (never), Output: 18566 (00:00:02 ago)
  LCP state: Opened
  NCP state: inet: Opened, inet6: Not-configured, iso: Not-configured, mpls:
  Not-configured
  CHAP state: Closed
  PAP state: Success
    Protocol inet, MTU: 1500
      Flags: Sendbcast-pkt-to-re
      Addresses, Flags: Is-Primary
        Local: 45.63.24.1
```

Meaning This output shows information about the PPPoE logical interface created on the underlying VLAN demux interface. The output includes the PPPoE family and aggregated Ethernet redundant link information, and shows input and output traffic for the PPPoE interface.

Related Documentation

- Subscriber Interfaces and Demultiplexing Overview on page 70
- Static or Dynamic Demux Subscriber Interfaces over Aggregated Ethernet Overview on page 89
- *Configuring Static Subscriber Interfaces Using VLAN Demux Interfaces*
- Configuring the PPPoE Family for an Underlying Interface on page 159

Example: Configuring a Dynamic PPPoE Subscriber Interface on a Static Underlying VLAN Demux Interface over Aggregated Ethernet

This example shows how you can configure dynamic PPPoE subscriber interfaces over aggregated Ethernet bundles to provide subscriber link redundancy.

Requirements

PPPoE over VLAN demux interfaces over aggregated Ethernet requires the following hardware and software:

- MX Series 3D Universal Edge Routers
- MPCs
- Junos OS Release 11.2 or later

No special configuration beyond device initialization is required before you can configure this feature.

Overview

Aggregated Ethernet bundles enable link redundancy between the router and networking devices connected by Ethernet links. This example describes how to configure link redundancy for dynamic PPPoE subscribers over aggregated Ethernet interface, **ae0**, with an intermediate static VLAN demux interface, **demux0.100**. Sample tasks include configuring a two-member aggregated Ethernet bundle, configuring a static VLAN demux interface that underlies the PPPoE subscriber interface, and configuring the dynamic profile that establishes the dynamic PPPoE subscriber interfaces.

The dynamic PPPoE profile (**pppoe-profile**) creates the PPPoE subscriber interface. It also configures the router to act as a PPPoE server and enables the local address to be derived from the specified address without assigning an explicit IP address to the interface. The **pppoe-profile** dynamic profile is assigned to the static, intermediate VLAN demux interface (**demux0.100**), which is configured with the PPPoE family (**family pppoe**) attributes. This dynamic profile includes the following predefined variables:

- **$junos-interface-unit**—Represents the logical unit number of the dynamic PPPoE logical interface. This predefined variable is dynamically replaced with the unit number supplied by the router when the subscriber logs in.
- **$junos-underlying-interface**—Represents the name of the underlying Ethernet interface. This predefined variable is dynamically replaced with the interface name supplied by the router when the subscriber logs in.

This example does not show all possible configuration choices.

Configuration

CLI Quick Configuration

To quickly configure link redundancy for dynamic PPPoE subscribers over a static VLAN demux interface over aggregated Ethernet, copy the following commands, paste them in a text file, remove any line breaks, and then copy and paste the commands into the CLI.

```
[edit]
set chassis aggregated-devices ethernet device-count 1
set interfaces ge-5/0/3 gigether-options 802.3ad ae0
set interfaces ge-5/0/3 gigether-options 802.3ad primary
set interfaces ge-5/1/2 gigether-options 802.3ad ae0
set interfaces ge-5/1/2 gigether-options 802.3ad backup
set interfaces ae0 flexible-vlan-tagging
set interfaces ae0 aggregated-ether-options link-protection
set interfaces demux0 unit 100 vlan-id 100
set interfaces demux0 unit 100 demux-options underlying-interface ae0
set interfaces demux0 unit 100 family pppoe access-concentrator pppoe-server-1
set interfaces demux0 unit 100 family pppoe duplicate-protection
set interfaces demux0 unit 100 family pppoe dynamic-profile pppoe-profile
edit dynamic-profiles pppoe-profile
edit interfaces pp0 unit $junos-interface-unit
set pppoe-options underlying-interface $junos-underlying-interface
set pppoe-options server
set family inet unnumbered-address lo0.0
top
```

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 link redundancy for dynamic PPPoE subscribers over a static VLAN demux interface over aggregated Ethernet:

1. Define the number of aggregated Ethernet devices on the router.

   ```
   [edit chassis]
   user@host# set aggregated-devices ethernet device-count 1
   ```

2. Configure a two-link aggregated Ethernet logical interface to serve as the underlying interface for the static VLAN demux subscriber interface. In this example, the LAG bundle is configured for one-to-one active/backup link redundancy. To support link redundancy at the MPC level, the LAG bundle attaches to ports from two different MPCs.

   ```
   [edit interfaces]
   user@host# set ge-5/0/3 gigether-options 802.3ad ae0
   user@host# set ge-5/0/3 gigether-options 802.3ad primary
   user@host# set ge-5/1/2 gigether-options 802.3ad ae0
   user@host# set ge-5/1/2 gigether-options 802.3ad backup
   ```

3. Enable link protection on the aggregated Ethernet logical interface and configure support for single and dual (stacked) VLAN tags.

   ```
   [edit interfaces]
   ```

user@host# **set ae0 aggregated-ether-options link-protection**
user@host# **set ae0 flexible-vlan-tagging**

4. Configure the VLAN demux interface over the aggregated Ethernet logical interface.

   [edit interfaces]
   user@host# **set demux0 unit 100 vlan-id 100**
   user@host# **set demux0 unit 100 demux-options underlying-interface ae0**

5. Configure the PPPoE family attributes on the VLAN demux interface, including the dynamic profile.

   [edit interfaces]
   user@host# **set demux0 unit 100 family pppoe access-concentrator pppoe-server-1**
   user@host# **set demux0 unit 100 family pppoe duplicate-protection**
   user@host# **set demux0 unit 100 family pppoe dynamic-profile pppoe-profile**

6. Configure the dynamic profile that creates the PPPoE subscriber interfaces.

   [edit dynamic-profiles pppoe-profile]
   user@host# **edit interfaces pp0 unit $junos-interface-unit**
   [edit dynamic-profiles pppoe-profile interfaces pp0 unit "$junos-interface-unit"]
   user@host# **set pppoe-options underlying-interface $junos-underlying-interface**
   user@host# **set pppoe-options server**
   user@host# **set family inet unnumbered-address lo0.0**

Results From configuration mode, confirm the aggregated device configuration by entering the **show chassis** command. Confirm the interface configuration by entering the **show interfaces** command. Confirm the dynamic profile configuration by entering the **show dynamic-profiles** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
user@host# show chassis
aggregated-devices {
  ethernet {
    device-count 1;
  }
}
```

```
[edit]
user@host# show interfaces
  ge-5/0/3 {
    gigether-options {
      802.3ad {
        ae0;
        primary;
      }
    }
  }
  ge-5/1/2 {
    gigether-options {
      802.3ad {
        ae0;
        backup;
      }
    }
```

```
    }
    ae0 {
        flexible-vlan-tagging;
        aggregated-ether-options {
            link-protection;
        }
    }
    demux0 {
        unit 100 {
            vlan-id 100;
            demux-options {
                underlying-interface ae0;
            }
            family pppoe {
                access-concentrator pppoe-server-1
                duplicate-protection;
                dynamic-profile pppoe-profile;
            }
        }
    }
[edit]
user@host# show dynamic-profiles
pppoe-profile {
    interfaces {
        pp0 {
            unit $junos-interface-unit {
                pppoe-options {
                    underlying-interface $junos-underlying-interface;
                    server;
                }
                family inet {
                    unnumbered-address lo0.0;
                }
            }
        }
    }
}
```

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

Verification

To confirm that the configuration is working properly, perform these tasks:

- Verifying the Aggregated Ethernet Interface Configuration on page 174
- Verifying the demux0 Interface Configuration on page 175

Verifying the Aggregated Ethernet Interface Configuration

Purpose Verify that the interface values match your configuration, the link is up, and traffic is flowing.

Action From operational mode, enter the **show interfaces redundancy** command.

```
user@host> show interfaces redundancy
Interface  State           Last change  Primary     Secondary    Current status
ae0        On primary                    ge-5/0/3     ge-5/1/2     both up
```

From operational mode, enter the **show interfaces ae0** command.

```
user@host> show interfaces ae0
Physical interface: ae0, Enabled, Physical link is Up
  Interface index: 128, SNMP ifIndex: 606
  Link-level type: Ethernet, MTU: 1522, Speed: 1Gbps, BPDU Error: None,
  MAC-REWRITE Error: None, Loopback: Disabled, Source filtering: Disabled,
  Flow control: Disabled, Minimum links needed: 1, Minimum bandwidth needed: 0
  Device flags   : Present Running
  Interface flags: SNMP-Traps Internal: 0x4000
  Current address: 00:1f:12:b8:ef:c0, Hardware address: 00:1f:12:b8:ef:c0
  Last flapped   : 2011-03-11 13:24:18 PST (2d 03:34 ago)
  Input rate     : 1984 bps (2 pps)
  Output rate    : 0 bps (0 pps)

  Logical interface ae0.32767 (Index 69) (SNMP ifIndex 709)
    Flags: SNMP-Traps 0x4004000 VLAN-Tag [ 0x0000.0 ]  Encapsulation: ENET2
    Statistics        Packets        pps         Bytes          bps
    Bundle:
        Input :        371259          2      46036116         1984
        Output:             0          0             0            0
    Protocol multiservice, MTU: Unlimited
      Flags: Is-Primary
```

Meaning The **show interfaces redundancy** output shows the redundant link configuration and that both link interfaces are up. The **show interfaces ae0** output shows that the aggregated Ethernet interface is up and that traffic is being received on the logical interface.

Verifying the demux0 Interface Configuration

Purpose Verify that the VLAN demux interface displays the configured PPPoE family attributes and the member links in the aggregated Ethernet bundle.

Action From operational mode, enter the **show interfaces demux0** command.

```
user@host> show interfaces demux0.100
Logical interface demux0.100 (Index 76) (SNMP ifIndex 61160)
    Flags: SNMP-Traps 0x4000 VLAN-Tag [ 0x8100.100 ]
    Encapsulation: ENET2
    Demux:
      Underlying interface: ae0 (Index 199)
    Link:
      ge-5/0/3
      ge-5/1/2
    Input packets : 2
    Output packets: 18575
    Protocol pppoe
      Dynamic Profile: pppoe-profile,
      Service Name Table: None,
      Max Sessions: 16000, Duplicate Protection: On,
      AC Name: pppoe-server-1
```

Alternatively, you can enter **show pppoe underlying-interfaces detail** to display the state and PPPoE family configuration for all configured underlying interfaces. The output also provides information about PPPoE negotiation on a per-VLAN basis.

Meaning The output shows the name of the underlying interface, the member links of the aggregated bundle, and the PPPoE family configuration. The output shows packet counts when traffic is present on the logical interface.

Related Documentation

- Subscriber Interfaces and Demultiplexing Overview on page 70
- Static or Dynamic Demux Subscriber Interfaces over Aggregated Ethernet Overview on page 89
- Configuring Dynamic Subscriber Interfaces Using VLAN Demux Interfaces in Dynamic Profiles on page 79
- Configuring the PPPoE Family for an Underlying Interface on page 159
- Configuring a Basic PPPoE Dynamic Profile on page 146

Example: Configuring a Dynamic PPPoE Subscriber Interface on a Dynamic Underlying VLAN Demux Interface over Aggregated Ethernet

This example shows how you can configure dynamic PPPoE subscriber interfaces over aggregated Ethernet bundles to provide subscriber link redundancy.

- Requirements on page 176
- Overview on page 176
- Configuration on page 177
- Verification on page 182

Requirements

PPPoE over VLAN demux interfaces over aggregated Ethernet requires the following hardware and software:

- MX Series 3D Universal Edge Routers
- MPCs
- Junos OS Release 11.2 or later

No special configuration beyond device initialization is required before you can configure this feature.

Overview

Aggregated Ethernet bundles enable link redundancy between the router and networking devices connected by Ethernet links. This example describes how to configure link redundancy for dynamic PPPoE subscribers over aggregated Ethernet with an intermediate dynamic VLAN demux interface. Sample tasks include configuring a two-member aggregated Ethernet bundle, configuring dynamic profiles that establish the dynamic

VLAN demux interface that underlies the PPPoE subscriber interface, and configuring the dynamic profile that establishes the dynamic PPPoE subscriber interfaces.

In this example, two different dynamic profiles are configured to instantiate either VLAN (**vlan-profile**) or S-VLAN (**svlan-profile**) demux interfaces. These profiles define PPPoE family options and include the dynamic PPPoE profile (**pppoe-profile**) that creates the PPPoE subscriber interface. Junos OS predefined variables are used in each profile to represent the interfaces and VLAN identifiers that are dynamically created. These dynamic profiles include the following predefined variables:

- **$junos-interface-unit**—Represents the logical unit number of the dynamic VLAN demux interface. This predefined variable is dynamically replaced with the unit number supplied by the router when the subscriber logs in.
- **$junos-interface-ifd-name**—Represents the underlying logical interface on which the PPPoE subscriber interface is created. This predefined variable is dynamically replaced with the name of the underlying interface supplied by the router when the subscriber logs in.
- **$junos-vlan-id**—Represents the VLAN identifier. This predefined variable is dynamically replaced with a VLAN ID when the subscriber logs in. The VLAN ID is allocated within the VLAN range specified in the aggregated Ethernet configuration. In the case of the S-VLAN demux, **$junos-vlan-id** represents the inner VLAN identifier.
- **$junos-stacked-vlan-id**—Represents the outer VLAN identifier for the stacked VLAN. This predefined variable is dynamically replaced with a VLAN ID when the subscriber logs in. The VLAN ID is allocated within the VLAN range specified in the aggregated Ethernet configuration. This variable is not used for the VLAN demux configuration.

The dynamic PPPoE profile (**pppoe-profile**) creates the PPPoE subscriber interface. It also configures the router to act as a PPPoE server and enables the local address to be derived from the specified address without assigning an explicit IP address to the interface. The **pppoe-profile** dynamic profile is assigned to the dynamic, intermediate VLAN and S-VLAN demux interfaces. This dynamic profile includes the following predefined variables:

- **$junos-interface-unit**—Represents the logical unit number of the dynamic PPPoE logical interface. This predefined variable is dynamically replaced with the unit number supplied by the router when the subscriber logs in.
- **$junos-underlying-interface**—Represents the name of the underlying Ethernet interface. This predefined variable is dynamically replaced with the interface name supplied by the router when the subscriber logs in.

This example does not show all possible configuration choices.

Configuration

CLI Quick Configuration

To quickly configure link redundancy for dynamic PPPoE subscribers over a dynamic VLAN demux interface over aggregated Ethernet, copy the following commands, paste them in a text file, remove any line breaks, and then copy and paste the commands into the CLI.

```
[edit]
set chassis aggregated-devices ethernet device-count 1
set interfaces ge-5/0/3 gigether-options 802.3ad ae0
set interfaces ge-5/0/3 gigether-options 802.3ad primary
set interfaces ge-5/1/2 gigether-options 802.3ad ae0
set interfaces ge-5/1/2 gigether-options 802.3ad backup
edit interfaces ae0
set flexible-vlan-tagging
set aggregated-ether-options link-protection
edit auto-configure
set vlan-ranges dynamic-profile vlan-profile accept pppoe
set vlan-ranges dynamic-profile vlan-profile ranges 1-4094
set stacked-vlan-ranges dynamic-profile svlan-profile accept pppoe
set stacked-vlan-ranges dynamic-profile svlan-profile ranges 1-4094,1-4094
top
edit dynamic-profiles pppoe-profile
edit interfaces pp0 unit $junos-interface-unit
set pppoe-options underlying-interface $junos-underlying-interface
set pppoe-options server
set family inet unnumbered-address lo0.0
top
edit dynamic-profiles vlan-profile interfaces demux0
edit unit $junos-interface-unit
set vlan-id $junos-vlan-id
set demux-options underlying-interface $junos-interface-ifd-name
set family pppoe access-concentrator pppoe-server-1
set family pppoe duplicate-protection
set family pppoe dynamic-profile pppoe-profile
top
edit dynamic-profiles svlan-profile interfaces demux0
edit unit $junos-interface-unit
set vlan-tags outer $junos-stacked-vlan-id
set vlan-tags inner $junos-vlan-id
set demux-options underlying-interface $junos-interface-ifd-name
set family pppoe access-concentrator pppoe-server-1
set family pppoe duplicate-protection
set family pppoe dynamic-profile pppoe-profile
top
```

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 link redundancy for dynamic PPPoE subscribers over a dynamic VLAN demux interface over aggregated Ethernet:

1. Define the number of aggregated Ethernet devices on the router.

   ```
   [edit chassis]
   user@host# set aggregated-devices ethernet device-count 1
   ```

2. Configure a two-link aggregated Ethernet logical interface to serve as the underlying interface for the dynamic VLAN demux subscriber interface. In this example, the LAG bundle is configured for one-to-one active/backup link redundancy. To support

link redundancy at the MPC level, the LAG bundle attaches to ports from two different MPCs.

```
[edit interfaces]
user@host# set ge-5/0/3 gigether-options 802.3ad ae0
user@host# set ge-5/0/3 gigether-options 802.3ad primary
user@host# set ge-5/1/2 gigether-options 802.3ad ae0
user@host# set ge-5/1/2 gigether-options 802.3ad backup
```

3. Enable link protection on the aggregated Ethernet logical interface and configure support for single and dual (stacked) VLAN tags.

```
[edit interfaces]
user@host# set ae0 aggregated-ether-options link-protection
user@host# set ae0 flexible-vlan-tagging
```

4. Configure the parameters for automatically configuring VLANs and S-VLANs, including the VLAN ranges and dynamic profiles.

```
[edit interfaces]
user@host# set ae0 auto-configure vlan-ranges dynamic-profile vlan-profile accept
  pppoe
user@host# set ae0 auto-configure vlan-ranges dynamic-profile vlan-profile ranges
  1-4094
user@host# set ae0 auto-configure stacked-vlan-ranges dynamic-profile
  svlan-profile accept pppoe
user@host# set ae0 auto-configure stacked-vlan-ranges dynamic-profile
  svlan-profile ranges 1-4094,1-4094
```

5. Configure the dynamic profile that creates the PPPoE subscriber interface.

```
[edit dynamic-profiles pppoe-profile]
user@host# edit interfaces pp0 unit $junos-interface-unit
[edit dynamic-profiles pppoe-profile interfaces pp0 unit "$junos-interface-unit"]
user@host# set pppoe-options underlying-interface $junos-underlying-interface
user@host# set pppoe-options server
user@host# set family inet unnumbered-address lo0.0
```

6. Configure the dynamic profile that creates VLAN demux underlying interfaces, including the PPPoE family attributes.

```
[edit dynamic-profiles vlan-profile]
user@host# edit interfaces demux0 unit $junos-interface-unit
[edit dynamic-profiles vlan-profile interfaces demux0 unit "$junos-interface-unit"]
user@host# set vlan-id $junos-vlan-id
user@host# set demux-options underlying-interface $junos-interface-ifd-name
user@host# set family pppoe access-concentrator pppoe-server-1
user@host# set family pppoe duplicate-protection
user@host# set family pppoe dynamic-profile pppoe-profile
```

7. Configure the dynamic profile that creates S-VLAN demux underlying interfaces, including the PPPoE family attributes.

```
[edit dynamic-profiles svlan-profile]
user@host# edit interfaces demux0 unit $junos-interface-unit
[edit dynamic-profiles svlan-profile interfaces demux0 unit "$junos-interface-unit"]
user@host# set vlan-tags outer $junos-stacked-vlan-id
user@host# set vlan-tags inner $junos-vlan-id
user@host# set demux-options underlying-interface $junos-interface-ifd-name
```



```
user@host# set family pppoe access-concentrator pppoe-server-1
user@host# set family pppoe duplicate-protection
user@host# set family pppoe dynamic-profile pppoe-profile
```

Results From configuration mode, confirm the aggregated device configuration by entering the **show chassis** command. Confirm the interface configuration by entering the **show interfaces** command. Confirm the dynamic profile configuration by entering the **show dynamic-profiles** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
user@host# show chassis
aggregated-devices {
  ethernet {
    device-count 1;
  }
}
```

```
[edit]
user@host# show interfaces
  ge-5/0/3 {
    gigether-options {
      802.3ad {
        ae0;
        primary;
      }
    }
  }
  ge-5/1/2 {
    gigether-options {
      802.3ad {
        ae0;
        backup;
      }
    }
  }
  ae0 {
    flexible-vlan-tagging;
    aggregated-ether-options {
      link-protection;
    }
    auto-configure {
      vlan-ranges {
        dynamic-profile {
          vlan-profile {
            accept pppoe;
            vlan-ranges 1–4094};
          }
        }
      }
      stacked-vlan-ranges {
        dynamic-profile {
          svlan-profile {
            accept pppoe;
            vlan-ranges 1–4094,1–4094;
```



```
            }
          }
        }
      }
    }

[edit]
user@host# show dynamic-profiles
pppoe-profile {
    interfaces {
      pp0 {
        unit $junos-interface-unit {
          pppoe-options {
            underlying-interface $junos-underlying-interface;
            server;
          }
          family inet {
            unnumbered-address lo0.0;
          }
        }
      }
    }
}
vlan-profile {
    interfaces {
      demux0 {
        unit "$junos-interface-unit" {
          vlan-id "$junos-vlan-id";
          demux-options {
            underlying-interface "$junos-interface-ifd-name";
          }
          family pppoe {
            access-concentrator pppoe-server-1;
            duplicate-protection;
            dynamic-profile pppoe-profile;
          }
        }
      }
    }
}
svlan-profile {
    interfaces {
      demux0 {
        unit "$junos-interface-unit" {
          vlan-tags outer "$junos-stacked-vlan-id" inner "$junos-vlan-id";
          demux-options {
            underlying-interface "$junos-interface-ifd-name";
          }
          family pppoe {
            access-concentrator pppoe-server-1;
            duplicate-protection;
            dynamic-profile pppoe-profile;
          }
        }
      }
    }
```

```
}
```

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

Verification

To confirm that the configuration is working properly, perform this task:

- Verifying the Aggregated Ethernet Interface Configuration on page 182

Verifying the Aggregated Ethernet Interface Configuration

Purpose Verify that the interface values match your configuration, the link is up, and traffic is flowing.

Action From operational mode, enter the **show interfaces redundancy** command.

```
user@host> show interfaces redundancy
Interface  State           Last change  Primary      Secondary    Current status
ae0        On primary                    ge-5/0/3     ge-5/1/2      both up
```

From operational mode, enter the **show interfaces ae0** command.

```
user@host> show interfaces ae0
Physical interface: ae0, Enabled, Physical link is Up
  Interface index: 128, SNMP ifIndex: 606
  Link-level type: Ethernet, MTU: 1522, Speed: 1Gbps, BPDU Error: None,
  MAC-REWRITE Error: None, Loopback: Disabled, Source filtering: Disabled,
  Flow control: Disabled, Minimum links needed: 1, Minimum bandwidth needed: 0
  Device flags   : Present Running
  Interface flags: SNMP-Traps Internal: 0x4000
  Current address: 00:1f:12:b8:ef:c0, Hardware address: 00:1f:12:b8:ef:c0
  Last flapped   : 2011-03-11 13:24:18 PST (2d 03:34 ago)
  Input rate     : 1984 bps (2 pps)
  Output rate    : 0 bps (0 pps)

  Logical interface ae0.32767 (Index 69) (SNMP ifIndex 709)
    Flags: SNMP-Traps 0x4004000 VLAN-Tag [ 0x0000.0 ]  Encapsulation: ENET2
    Statistics        Packets        pps         Bytes          bps
    Bundle:
        Input :        371259          2      46036116         1984
        Output:             0          0             0            0
    Protocol multiservice, MTU: Unlimited
      Flags: Is-Primary
```

Meaning The **show interfaces redundancy** output shows the redundant link configuration and that both link interfaces are up. The **show interfaces ae0** output shows that the aggregated Ethernet interface is up and that traffic is being received on the logical interface.

Related Documentation

- Subscriber Interfaces and Demultiplexing Overview on page 70
- Static or Dynamic Demux Subscriber Interfaces over Aggregated Ethernet Overview on page 89
- Configuring Dynamic Subscriber Interfaces Using VLAN Demux Interfaces in Dynamic Profiles on page 79

- Configuring the PPPoE Family for an Underlying Interface on page 159
- Configuring a Basic PPPoE Dynamic Profile on page 146

CHAPTER 18

Configuring PPPoE Sessions for Subscriber Interfaces

PPPoE Maximum Session Limit Overview

The maximum session limit for PPPoE subscriber interfaces specifies the maximum number of concurrent static or dynamic PPPoE logical interfaces (sessions) that the router can activate on the PPPoE underlying interface, or the maximum number of active static or dynamic PPPoE sessions that the router can establish with a particular service entry in a PPPoE service name table.

You can configure the PPPoE maximum session limit in one of two ways:

- On a per-interface basis, by using the **max-sessions** CLI statement
- (Default) On a per-subscriber basis, by using the value returned by RADIUS in the Max-Clients-Per-Interface Juniper Networks vendor-specific attribute (VSA) [26-143]

This overview describes the concepts you need to understand to configure the PPPoE maximum session limit, and covers the following topics:

Per-Interface Configuration for PPPoE Maximum Session Limit Using the CLI

To configure the PPPoE maximum session limit for a particular interface, you can use the **max-sessions** statement to specify either or both of the following, depending on the hierarchy level at which you include the statement:

- The maximum number of concurrent PPPoE sessions that the router can activate on the PPPoE underlying interface

- The maximum number of active PPPoE sessions using either static or dynamic PPPoE interfaces that the router can establish with a particular named service entry, **empty** service entry, or **any** service entry in a PPPoE service name table

You can configure the PPPoE maximum session value from 1 through the platform-specific default for your router. The default value is equal to the maximum number of PPPoE sessions supported on your routing platform. If the number of active PPPoE sessions exceeds the value configured with the **max-sessions** statement, the router prohibits creation of any new PPPoE sessions, and the PPPoE application on the router returns a PPPoE Active Discovery Session (PADS) packet with an error to the PPPoE client.

Changing the PPPoE maximum session value has no effect on dynamic PPPoE subscriber interfaces that are already active.

Per-Subscriber Configuration for PPPoE Maximum Session Limit Using RADIUS

To configure the PPPoE maximum session limit for a particular subscriber, you can use the value returned by the RADIUS server in the Max-Clients-Per-Interface Juniper Networks VSA [26-143] during the subscriber authentication process. For PPPoE clients, the Max-Clients-Per-Interface VSA returns the maximum number of sessions (PPPoE subinterfaces) per PPPoE major interface.

By default, the PPPoE maximum session value returned by RADIUS in the Max-Clients-Per-Interface VSA takes precedence over the PPPoE maximum session value configured with the **max-sessions** statement.

If you configure multiple subscribers on the same PPPoE underlying VLAN interface and RADIUS returns a different PPPoE maximum session value for each subscriber, the router uses the most recent PPPoE maximum session value returned by RADIUS to determine whether to override the current PPPoE maximum session value and create the new PPPoE session.

The following sequence describes how the router obtains the PPPoE maximum session value from RADIUS when a PPPoE subscriber logs in to initiate a session with the router. (In a PPPoE subscriber network, the router functions as a *remote access concentrator*, also known as a *PPPoE server*.)

1. The PPPoE client and the router participate in the PPPoE Discovery process to establish the PPPoE connection.
2. The PPP Link Control Protocol (LCP) negotiates the PPP link between the client and the router.
3. The PPP application sends the subscriber authentication request to the AAA application.
4. AAA sends the authentication request to an external RADIUS server.
5. The RADIUS server returns the PPPoE maximum session value for that subscriber to AAA in the Max-Clients-Per-Interface VSA as part of an Access-Accept message.

NOTE: The RADIUS server does not return the Max-Clients-Per-Interface VSA in Change of Authorization Request (CoA-Request) messages.

6. AAA passes the response from RADIUS to PPP.
7. PPP validates the subscriber parameters and, if authentication succeeds, passes the PPPoE maximum session value returned by RADIUS to the PPPoE application.
8. PPPoE uses the maximum session value returned by RADIUS to determine whether to override the current PPPoE maximum session value and create or tear down the new PPPoE session.

Override of PPPoE Maximum Session Limit from RADIUS

By default, the PPPoE maximum session value returned by RADIUS in the Max-Clients-Per-Interface VSA [26-143] takes precedence over the PPPoE maximum session value configured with the **max-sessions** statement. To configure the router to ignore (clear) the PPPoE maximum session value returned by the RADIUS server in the Max-Clients-Per-Interface VSA, include the **max-sessions-vsa-ignore** statement at the same hierarchy levels that you can specify the **max-sessions** statement.

Including the **max-sessions-vsa-ignore** statement in your configuration restores the PPPoE maximum session value on the underlying interface to the value configured in the CLI with the **max-sessions** statement.

Related Documentation

- Guidelines for Using PPPoE Maximum Session Limit from RADIUS on page 187
- *Juniper Networks VSAs Supported by the AAA Service Framework*
- Limiting the Maximum Number of PPPoE Sessions on the Underlying Interface on page 189
- Dynamic PPPoE Subscriber Interfaces over Static Underlying Interfaces Overview on page 153

Guidelines for Using PPPoE Maximum Session Limit from RADIUS

Consider the following guidelines when you use the PPPoE maximum session value returned by RADIUS in the Max-Clients-Per-Interface vendor-specific attribute (VSA) [26-143]:

- If the current number of sessions (including newly created sessions) is *less than* the new PPPoE maximum session value returned by RADIUS, the PPPoE application overrides the current value and enables interface creation to proceed.
- If the current number of sessions (including newly created sessions) is *equal to* the new PPPoE maximum session value returned by RADIUS, the PPPoE application overrides the current value and enables interface creation to proceed.

- If the current number of sessions (including newly created sessions) is *greater than* the new PPPoE maximum session value returned by RADIUS, the PPPoE application overrides the current value and brings down the new interface.

To illustrate these guidelines, Table 6 on page 188 shows examples of how the router handles the PPPoE session when the current number of sessions is less than (first row), equal to (second row), and greater than (third row) the new PPPoE maximum session value returned by RADIUS when a new subscriber logs in.

Table 6: Sample PPPoE Maximum Session Values During Subscriber Login

New PPPoE Maximum Session Value from RADIUS	Current PPPoE Maximum Session Value	Existing Number of PPPoE Sessions	New PPPoE Maximum Session Value	New Number of PPPoE Sessions	Status of Session
10	5	4	10	5	PPPoE session up
5	5	4	5	5	PPPoE session up
3	5	4	3	4	PPPoE session down

Related Documentation

- PPPoE Maximum Session Limit Overview on page 185
- *Juniper Networks VSAs Supported by the AAA Service Framework*
- Limiting the Maximum Number of PPPoE Sessions on the Underlying Interface on page 189
- Dynamic PPPoE Subscriber Interfaces over Static Underlying Interfaces Overview on page 153

Limiting the Maximum Number of PPPoE Sessions on the Underlying Interface

You can limit the maximum number of concurrent static or dynamic PPPoE logical interfaces (sessions) that the router can activate on the PPPoE underlying interface, or the maximum number of active static or dynamic PPPoE sessions that the router can establish with a particular service entry in a PPPoE service name table.

You can configure the PPPoE maximum session limit in either of the following ways:

- On a per-interface basis, by using the **max-sessions** CLI statement.
- (Default) On a per-subscriber basis, by using the value returned by RADIUS in the Max-Clients-Per-Interface Juniper Networks vendor-specific attribute (VSA) [26-143]. By default, the PPPoE maximum session value returned by RADIUS in the Max-Clients-Per-Interface VSA takes precedence over the PPPoE maximum session value configured with the **max-sessions** statement.

Optionally, you can configure the router to ignore the PPPoE maximum session value returned by RADIUS in the Max-Clients-Per-Interface VSA by including the **max-sessions-vsa-ignore** statement in your configuration. Using the **max-sessions-vsa-ignore** statement restores the PPPoE maximum session value on the underlying interface to the value configured in the CLI with the **max-sessions** statement.

Before you begin:

- Configure the PPPoE underlying interface.

  To configure the underlying interface for use with a PPPoE dynamic profile, see "Configuring an Underlying Interface for Dynamic PPPoE Subscriber Interfaces" on page 156.

  To configure the PPPoE family for an underlying interface, see "Configuring the PPPoE Family for an Underlying Interface" on page 159.

To configure the PPPoE maximum session limit:

1. Specify that you want to configure PPPoE-specific options on the underlying interface:
   - For a PPPoE family in a dynamic profile for a VLAN demultiplexing (demux) logical interface:

     [edit dynamic-profiles *profile-name* interfaces demux0 unit *logical-unit-number*]
     user@host# **edit family pppoe**

   - For a PPPoE family in a dynamic profile:

     [edit dynamic-profiles *profile-name* interfaces *interface-name* unit *logical-unit-number*]
     user@host# **edit family pppoe**

   - For a PPPoE underlying interface in a dynamic profile:

     [edit dynamic-profiles *profile-name* interfaces *interface-name* unit *logical-unit-number*]
     user@host# **edit pppoe-underlying-options**

   - For a PPPoE family on an underlying interface:

[edit interfaces *interface-name* unit *logical-unit-number*]
user@host# **edit family pppoe**

- For an underlying interface with PPPoE encapsulation:

  [edit interfaces *interface-name* unit *logical-unit-number*]
  user@host# **edit pppoe-underlying-options**

- For an underlying interface established with a particular service entry in a PPPoE service name table:

  [edit protocols pppoe service-name-tables *table-name*]
  user@host# **edit service *service-name***

2. Configure the maximum number of concurrent PPPoE sessions that the router can activate on the underlying interface in either of the following ways:

   - To configure the maximum number of concurrent PPPoE sessions on a per-interface basis, from 1 to the platform-specific default for your router, use the **max-sessions** statement:

     [edit interfaces *interface-name* unit *logical-unit-number* pppoe-underlying-options]
     user@host# **set max-sessions *number***

   - To configure the maximum number of concurrent PPPoE sessions on a per-subscriber basis, use the value returned by RADIUS in the Max-Clients-Per-Interface Juniper Networks vendor-specific attribute (VSA) [26-143]. By default, the PPPoE maximum session value returned by RADIUS in the Max-Clients-Per-Interface VSA takes precedence over the PPPoE maximum session value configured with the **max-sessions** statement.

3. (Optional) To restore the PPPoE maximum session value on the underlying interface to the value configured in the CLI with the **max-sessions** statement, configure the router to ignore the value returned by RADIUS in the Max-Clients-Per-Interface VSA.

   [edit interfaces *interface-name* unit *logical-unit-number* pppoe-underlying-options]
   user@host# **set max-sessions-vsa-ignore**

   NOTE: You can issue the **max-sessions-vsa-ignore** statement at the same hierarchy levels as the **max-sessions** statement, with the exception of the **[edit protocols pppoe service-name-tables *table-name* service *service-name*]** hierarchy level.

Related Documentation

- PPPoE Maximum Session Limit Overview on page 185
- Guidelines for Using PPPoE Maximum Session Limit from RADIUS on page 187
- *Juniper Networks VSAs Supported by the AAA Service Framework*
- Configuring an Underlying Interface for Dynamic PPPoE Subscriber Interfaces on page 156
- Configuring the PPPoE Family for an Underlying Interface on page 159
- Dynamic PPPoE Subscriber Interfaces over Static Underlying Interfaces Overview on page 153

CHAPTER 19

Configuring PPPoE Subscriber Session Lockout

PPPoE Subscriber Session Lockout Overview

PPPoE subscriber session lockout, which is sometimes referred to as PPPoE encapsulation type lockout, configures the router to temporarily prevent (lock out) a failed or short-lived static or dynamic PPPoE subscriber session from reconnecting for a default or configurable period of time. This time period, known as the *lockout period*, is derived from a formula and increases exponentially based on the number of successive reconnection failures.

This overview describes the concepts you need to understand to configure PPPoE subscriber session lockout, and covers the following topics:

Benefits of Using PPPoE Subscriber Session Lockout

Configuring and using PPPoE subscriber session lockout provides the following benefits:

- Reduces excessive loading on the router

By temporarily locking out failed or short-lived PPPoE sessions, PPPoE subscriber session lockout protects the router from excessive loading by:

  - Reducing the resources required to receive and process PPPoE control packets to negotiate and terminate short-lived connections
  - Reducing the resources required to allocate and deallocate services, such as class of service (CoS) and firewall filters, for failed or short-lived subscriber sessions

  PPPoE subscriber session lockout increases router efficiency by temporarily deferring failed or short-lived subscriber sessions in favor of those sessions that can complete successfully.

- Reduces excessive loading on external authentication, authorization, and accounting (AAA) servers

  PPPoE subscriber session lockout protects any external AAA servers, such as RADIUS or Diameter, from excessive loading:

  - As a result of failed or short-lived PPPoE subscriber sessions that occur repeatedly for the same subscriber
  - By reducing the resources required to authenticate and terminate these connections

- Enables lockout of a single failed or short-lived PPP session without disrupting other PPP sessions on the same PPPoE underlying interface

  In some subscriber network configurations, the PPPoE underlying interface supports multiple upper-layer PPP sessions. Because PPPoE subscriber session lockout identifies each subscriber session by its unique media access control (MAC) source address on the underlying interface, the router is able to lock out only the offending PPP session while enabling other PPP sessions on the same underlying interface to successfully negotiate the connection.

Supported Platforms and Underlying Interfaces for PPPoE Subscriber Session Lockout

You can configure PPPoE subscriber session lockout on the following platforms and underlying interface types:

- Supported platforms:
  - Intelligent Queuing 2 (IQ2) PICs on M120 Multiservice Edge Router and M320 Multiservice Edge Router
  - MPC/MIC interfaces on MX Series 3D Universal Edge Routers
- Supported PPPoE underlying interfaces:
  - Static VLAN logical interface
  - Static VLAN demultiplexing (demux) logical interface
  - Dynamic VLAN logical interface
  - Dynamic VLAN demux logical interface

How PPPoE Subscriber Session Lockout Works

PPPoE subscriber session lockout is disabled on the router by default. When you enable PPPoE subscriber session lockout by issuing the **short-cycle-protection** statement, the router does the following:

1. Detects a short-lived subscriber session, also referred to as a *short-cycle event*.

   A short-lived subscriber session is detected, partially or completely created, and terminated by the router within 150 seconds. The router identifies each PPPoE subscriber session by its unique MAC source address on the PPPoE underlying interface.

2. Tracks the time between repeated short-cycle events to determine whether to increase the lockout time for a subsequent short-cycle event.

3. Applies a time penalty for each short-cycle event based on a default or configured lockout period and the number of consecutive short-cycle events that occur repeatedly for the same subscriber.

   If you enable PPPoE subscriber session lockout but do not configure a lockout time range, the router uses the default lockout time range of 1 through 300 seconds (5 minutes).

4. Temporarily locks out the specified PPPoE subscriber by preventing connection to the router.

   During lockout, the router drops negotiation packets for the PPPoE subscriber session until the lockout period expires. When the lockout period expires, the PPPoE subscriber session and its associated MAC source address resume normal negotiation of the connection.

Repeated creation of multiple short-lived (short-cycle) PPPoE subscriber sessions can cause excessive loading on the router. Conditions that can cause a short-lived subscriber session include:

- Authentication denials from external AAA servers, such as RADIUS, due to the absence of a corresponding entry in the RADIUS database or due to improper login attempts
- Configuration errors within a dynamic profile or RADIUS record
- Insufficient memory resources to create a dynamic PPPoE subscriber interface
- Protocol failure or error within the dynamic PPPoE subscriber interface
- Client logout shortly after a successful login; this action creates a complete dynamic PPPoE subscriber interface before the interface is torn down

PPPoE Subscriber Session Lockout Period

The lockout period is the time during which the router temporarily prevents (locks out) a failed or short-lived PPPoE subscriber session identified by a unique MAC source address or an agent circuit identifier (ACI) value on the PPPoE underlying interface from reconnecting to the router. You can use the default lockout time range of 1 through 300 seconds (5 minutes), or you can override the default lockout period by configuring a nondefault lockout time in the range 1 through 86,400 seconds (24 hours).

PPPoE Encapsulation Type Lockout

Configuring and using PPPoE subscriber session lockout increases router efficiency and protects the router and any external AAA servers from excessive loading by temporarily deferring failed or short-lived subscriber sessions in favor of those sessions that can complete successfully. PPPoE subscriber session lockout enables you to prevent (lock out) a failed or short-lived PPPoE subscriber session from reconnecting to the router for a default or configurable period of time, based on either of the following options:

- The subscriber session's unique MAC source address on the PPPoE underlying interface

  This option, which is the default, locks out the offending PPPoE subscriber session identified by its MAC source address on the underlying interface.

- The ACI string contained in the DSL Forum Agent-Circuit-ID VSA [26-1] (option 0x105) of PPPoE Active Discovery Initiation (PADI) and PPPoE Active Discovery Request (PADR) control packets

  This option locks out all PPPoE subscriber sessions on the underlying interface that come from the same household and share the same ACI string in their PPPoE PADI and PADR control packets.

PPPoE subscriber session lockout based on the ACI value is particularly useful for configurations such as the following in which MAC source addresses are not unique on the PPPoE underlying interface:

- PPPoE interworking function sessions in which the MAC addresses of all PPPoE inter-working function sessions contain the MAC address of the DSLAM device
- Configurations in which the access node (usually a DSLAM device) overwrites the MAC source address in PPPoE packets received from the customer premises equipment (CPE) with its own MAC address for security purposes
- Duplicate MAC source addresses across disparate households in an N:1 (service VLAN) configuration, which requires the router to use a combination of the MAC source address and the ACI value to uniquely identify a subscriber

To configure temporary PPPoE subscriber session lockout based on the ACI value, include the short-cycle-protection statement with the new filter aci option for PPPoE subscriber sessions on any of the following underlying logical interfaces types:

- Dynamic or static VLAN interfaces (in the pppoe-underlying-options stanza)
- Dynamic or static VLAN demultiplexing (demux) interfaces (in the family pppoe stanza)

For example, the following statement configures temporary lockout based on ACI information for PPPoE subscriber sessions on a dynamic VLAN underlying interface. This statement specifies a nondefault lockout time in the range 20 through 120 seconds.

```
[edit dynamic-profiles my-vlan-profile interfaces "$junos-interface-ifd-name" unit
  "$junos-interface-unit" pppoe-underlying options]
user@host# set short-cycle-protection lockout-time-min 20 lockout-time-max 120 filter
  aci
```


The following statement configures temporary lockout based on ACI information for PPPoE subscriber sessions on a dynamic VLAN demux underlying interface. This statement uses the default lockout time range 1 through 300 seconds.

```
[edit dynamic-profiles my-demux-vlan-profile interfaces demux0 unit
  "$junos-interface-unit" family pppoe]
user@host# set short-cycle-protection filter aci
```

NOTE: MAC-based encapsulation type lockout is used if ACI is not in the PPPoE attributes when ACI-level encapsulation type lockout is configured. With ACI-based encapsulation type lockout, PPPoE clients without an ACI attribute are also locked-out.

PPPoE Subscriber Session Lockout and Duplicate Protection

Duplicate protection, which is disabled on the router by default, prevents the activation of another PPPoE subscriber session on the same PPPoE underlying interface when a PPPoE subscriber session with the same media access control (MAC) address is already active on that interface. When you configure PPPoE subscriber session lockout, we recommend that you enable duplicate protection to ensure that the MAC source address for each active PPPoE session is unique on the underlying interface.

With PPPoE subscriber session lockout configured, the router identifies subscriber sessions by their unique MAC source address. If the router detects a short-lived (short-cycle) subscriber session, it applies the default or configured lockout period to that MAC source address to temporarily prevent reconnection. If the MAC source address is not unique on the underlying interface, multiple PPPoE subscriber sessions with the same MAC source address might also be affected by the lockout.

PPPoE Subscriber Session Lockout and Automatic Removal of Dynamic Subscriber VLANs

You can configure automatic removal of subscriber VLANs that have no PPPoE client sessions by issuing the **remove-when-no-subscribers** statement at the **[edit interfaces *interface-name* auto-configure]** hierarchy level. If PPPoE subscriber session lockout is also configured, the router does not remove the unused subscriber VLAN until the lockout time has expired for each PPPoE client undergoing lockout on the underlying interface.

Related Documentation

- Understanding the Lockout Period for PPPoE Subscriber Session Lockout on page 196
- Configuring Lockout of PPPoE Subscriber Sessions on page 198
- Clearing Lockout of PPPoE Subscriber Sessions on page 199
- Verifying and Managing Dynamic PPPoE Configuration on page 211
- Dynamic PPPoE Subscriber Interfaces over Static Underlying Interfaces Overview on page 153

Understanding the Lockout Period for PPPoE Subscriber Session Lockout

When you configure PPPoE subscriber session lockout, the router applies a time penalty called the *lockout period* for each failed or short-lived subscriber session. During the lockout period, the router temporarily prevents (locks out) a failed or short-lived PPPoE subscriber session identified by a unique media access control (MAC) source address from reconnecting to the router.

This overview describes how the router determines and applies the PPPoE subscriber session lockout period, and covers the following topics:

Duration of PPPoE Subscriber Session Lockout Period

The duration of the lockout period is based on a default or configured lockout time and the number of consecutive short-cycle (short-lived) events that occur repeatedly for the same subscriber. When you include the **short-cycle-protection** statement to configure PPPoE subscriber session lockout on a PPPoE underlying interface, you can use the default lockout time range of 1 through 300 seconds (5 minutes), or you can override the default lockout period by configuring a nondefault lockout time in the range 1 through 86,400 seconds (24 hours).

The lockout time penalty applied by the router for each short-cycle event differs depending on the event. For example, some short-cycle events represent normal subscriber behavior, such as a PPPoE subscriber logging in once per hour to check e-mail and logging out shortly thereafter. The router does not noticeably penalize a subscriber for these types of events.

By contrast, other short-cycle events are the result of repeated attempts to log in to the router for reasons such as an incorrectly typed password, customer premises equipment (CPE) that performs repeated auto-retries, or malicious attempts to access the Internet illegally. For these types of short-cycle events, the router applies a lockout time penalty that starts with a short time interval and increases exponentially. In these instances, the initial lockout time is short enough to avoid noticeably penalizing a subscriber who, for example, types a password incorrectly several times before entering the correct one.

For example, using the default lockout time range of 1 through 300 seconds, the increasing lockout period on the router is: 1 second, 2 seconds, 4 seconds, 8 seconds, 16 seconds, 32 seconds, 64 seconds, 128 seconds, 256 seconds, and finally, 300 seconds (5 minutes).

How the Router Determines the PPPoE Subscriber Session Lockout Period

The router uses the following rules to determine the PPPoE subscriber session lockout period for short-lived PPPoE subscriber sessions:

- The lockout period is derived from the following formula:

  (minimum lockout time) * (2 ^ *n*-1)

  where *n* represents the number of consecutive short-cycle events for the same subscriber. The router identifies a PPPoE subscriber session by its MAC source address, which should be unique on the underlying PPPoE interface.

- The router increments the value of *n* when the time between short-cycle events is either within 15 minutes or the maximum lockout time, whichever is greater.

- When the time between short-cycle events is greater than either 15 minutes or the maximum lockout time, the value of *n* reverts to 1. This condition is referred to as a *lockout grace period*.

- The lockout period never exceeds the maximum configured lockout time.

  For example, for a configured (nondefault) lockout time in the range 20 through 120 seconds, the increasing lockout period on the router is: 20 seconds, 40 seconds, 80 seconds, and finally, 120 seconds (2 minutes).

- A *short-cycle event* is detected, partially or completely created, and terminated by the router within 150 seconds. The router tracks the time between short-cycle events to determine whether to increase the lockout time for a subsequent short-cycle event for the same subscriber.

  NOTE: **When the calculated lockout time is equal to or exceeds the maximum lockout time, the router uses the maximum lockout time value until the time to the next short-cycle event exceeds the greater of 15 minutes or the maximum lockout time value. At that point, the lockout time reverts to the minimum lockout time value.**

- The minimum lockout time value cannot exceed the maximum lockout time value.

  When the minimum and maximum lockout time values are equal, the lockout time becomes fixed at that value.

Related Documentation

- PPPoE Subscriber Session Lockout Overview on page 191
- Configuring Lockout of PPPoE Subscriber Sessions on page 198
- Clearing Lockout of PPPoE Subscriber Sessions on page 199
- Verifying and Managing Dynamic PPPoE Configuration on page 211
- Dynamic PPPoE Subscriber Interfaces over Static Underlying Interfaces Overview on page 153

Configuring Lockout of PPPoE Subscriber Sessions

You can configure the router to temporarily prevent (lock out) a failed or short-lived PPPoE subscriber session from reconnecting to the router for a default or configurable period of time. Configuring a lockout period on the PPPoE underlying interface for static or dynamic PPPoE subscriber sessions protects the router and any external authentication, authorization, and accounting (AAA) servers, such as RADIUS, from excessive loading as a result of failed or short-lived (also known as short-cycle) PPPoE subscriber sessions that occur repeatedly for the same subscriber.

You can configure the router to use the default PPPoE lockout period, 1 through 300 seconds (5 minutes). Alternatively, you can override the default lockout period by specifying a minimum lockout time and maximum lockout time, each of which can be from 1 through 86,400 seconds (24 hours).

Before you begin:

- Configure the PPPoE underlying interface.

  To configure the underlying interface for use with a PPPoE dynamic profile, see "Configuring an Underlying Interface for Dynamic PPPoE Subscriber Interfaces" on page 156.

  To configure the PPPoE family for an underlying interface, see "Configuring the PPPoE Family for an Underlying Interface" on page 159.

To configure temporary lockout of PPPoE subscriber sessions:

1. Specify that you want to configure PPPoE-specific options on the underlying interface:
   - For a PPPoE family in a dynamic profile for a VLAN demultiplexing (demux) logical interface:

     [edit dynamic-profiles *profile-name* interfaces demux0 unit *logical-unit-number*]
     user@host# **edit family pppoe**

   - For a PPPoE family in a dynamic profile:

     [edit dynamic-profiles *profile-name* interfaces *interface-name* unit *logical-unit-number*]
     user@host# **edit family pppoe**

   - For a PPPoE underlying interface in a dynamic profile:

     [edit dynamic-profiles *profile-name* interfaces *interface-name* unit *logical-unit-number*]
     user@host# **edit pppoe-underlying-options**

   - For a PPPoE family on an underlying interface:

     [edit interfaces *interface-name* unit *logical-unit-number*]
     user@host# **edit family pppoe**

   - For an underlying interface with PPPoE encapsulation:

     [edit interfaces *interface-name* unit *logical-unit-number*]
     user@host# **edit pppoe-underlying-options**

2. Enable duplicate protection to prevent negotiation of a dynamic or static PPPoE client session on the same underlying interface when a PPPoE client session with the same media access control (MAC) source address is already active on that interface.

   [edit interfaces *interface-name* unit *logical-unit-number* pppoe-underlying-options]
   user@host# set duplicate-protection

   BEST PRACTICE: **When you configure PPPoE subscriber session lockout, we recommend that you enable duplicate protection to ensure that the MAC source address for each PPPoE session is unique on the underlying interface.**

3. Enable PPPoE subscriber session lockout on the PPPoE underlying interface in either of the following ways:

   - To configure PPPoE subscriber session lockout with the default lockout period:

     [edit interfaces *interface-name* unit *logical-unit-number* pppoe-underlying-options]
     user@host# set short-cycle-protection

   - To configure PPPoE subscriber session lockout with a nondefault lockout period:

     [edit interfaces *interface-name* unit *logical-unit-number* pppoe-underlying-options]
     user@host# set short-cycle-protection lockout-time-min *minimum-seconds*
       lockout-time-max *maximum-seconds*

Related Documentation

- PPPoE Subscriber Session Lockout Overview on page 191
- Clearing Lockout of PPPoE Subscriber Sessions on page 199
- Configuring an Underlying Interface for Dynamic PPPoE Subscriber Interfaces on page 156
- Configuring the PPPoE Family for an Underlying Interface on page 159
- Dynamic PPPoE Subscriber Interfaces over Static Underlying Interfaces Overview on page 153

Clearing Lockout of PPPoE Subscriber Sessions

Purpose Clear the lockout condition for the PPPoE subscriber session associated with a unique MAC source address.

Action

- To clear the lockout condition for PPPoE subscriber sessions associated with all MAC source addresses on all underlying interfaces:

  user@host> clear pppoe lockout

- To clear the lockout condition for the PPPoE subscriber session associated with the specified MAC source address:

  user@host> clear pppoe lockout mac-address *mac-address*

- To clear the lockout condition for all PPPoE subscriber sessions on the specified underlying interface:

user@host> **clear pppoe lockout underlying-interfaces *underlying-interface-name***

- To clear the lockout condition for the PPPoE subscriber session associated with the specified MAC source address on the specified underlying interface:

  user@host> **clear pppoe lockout mac-address *mac-address* underlying-interfaces *underlying-interface-name***

- To verify that the lockout condition has been cleared:

  user@host> **show pppoe lockout**

Related Documentation

- PPPoE Subscriber Session Lockout Overview on page 191
- Configuring Lockout of PPPoE Subscriber Sessions on page 198
- Verifying and Managing Dynamic PPPoE Configuration on page 211
- CLI Explorer

CHAPTER 20

Configuring PPPoE Service Name Tables

Understanding PPPoE Service Name Tables

On an M120 router, M320 router, or MX Series router acting as a remote access concentrator (AC), also referred to as a *PPPoE server*, you can configure up to 32 PPPoE service name tables and assign the service name tables to PPPoE underlying interfaces. A *PPPoE service name table* defines the set of *services* that the router can provide to a PPPoE client. Service entries configured in a PPPoE service name table represent the *service name tags* transmitted between the client and the router in a PPPoE control packet.

This overview covers the following topics to help you understand and configure PPPoE service name tables:

Interaction Among PPPoE Clients and Routers During the Discovery Stage

In networks with mesh topologies, PPPoE clients are often connected to multiple PPPoE servers (remote ACs). During the PPPoE discovery stage, a PPPoE client identifies the Ethernet MAC address of the remote AC that can service its request, and establishes a unique PPPoE session identifier for a connection to that AC.

The following steps describe, at a high level, how the PPPoE client and the remote AC (router) use the PPPoE service name table to interact during the PPPoE discovery stage:

1. The PPPoE client broadcasts a PPPoE Active Discovery Initiation (PADI) control packet to all remote ACs in the network to request that an AC support certain services.

   The PADI packet must contain either, but not both, of the following:

   - One and only one nonzero-length service name tag that represents a specific client service
   - One and only one empty (zero-length) service name tag that represents an unspecified service

2. One or more remote ACs respond to the PADI packet by sending a PPPoE Active Discovery Offer (PADO) packet to the client, indicating that the AC can service the client request.

   To determine whether it can service a particular client request, the router matches the service name tag received in the PADI packet against the service name tags configured in its service name table. If a matching service name tag is found in the PPPoE service name table, the router sends the client a PADO packet that includes the name of the AC from which it was sent. If no matching service name tag is found in the PPPoE service name table, the router drops the PADI request and does not send a PADO response to the client.

3. The PPPoE client sends a unicast PPPoE Active Discovery Request (PADR) packet to the AC to which it wants to connect, based on the responses received in the PADO packets.
4. The selected AC sends a PPPoE Active Discovery Session (PADS) packet to establish the PPPoE connection with the client.

Service Entries and Actions in PPPoE Service Name Tables

A PPPoE service name table can include three types of service entries: named services, an **empty** service, and an **any** service. For each service entry, you specify the action to be taken by the underlying interface when the router receives a PADI packet containing the specified service name tag.

You can configure the following services and actions in a PPPoE service name table:

- Named service—Specifies a PPPoE client service that an AC can support. For example, you might configure named services associated with different subscribers who log in to the PPPoE server, such as **user1-service** or **user2-service**, or that correspond to different ISP service level agreements, such as **premium** and **standard**. Each PPPoE service name table can include a maximum of 512 named service entries, excluding **empty** and **any** service entries. A named service is associated with the **terminate** action by default.
- **empty** service—A service tag of zero length that represents an unspecified service. Each PPPoE service name table includes one empty service. The **empty** service is associated with the **terminate** action by default.

- **any** service—Acts as a default service for non-empty service entries that do not match the named service entries or **empty** service entry configured in the PPPoE service name table. Each PPPoE service name table includes one **any** service. The **any** service is useful when you want to match the agent circuit identifier and agent remote identifier information for a PPPoE client, but do not care about the contents of the service name tag transmitted in the control packet. The **any** service is associated with the **drop** action by default.
- Action—Specifies the action taken by the underlying PPPoE interface assigned to the PPPoE service name table on receipt of a PADI packet from the client containing a particular service request. You can configure one of the following actions for the associated named service, **empty** service, **any** service, or agent circuit identifier/agent remote identifier (ACI/ARI) pair in the PPPoE service name table on the router:
  - **terminate**—(Default) Directs the router to immediately respond to the PADI packet by sending the client a PADO packet containing the name of the AC that can service the request. Named services, **empty** services, and ACI/ARI pairs are associated with the **terminate** action by default. Configuring the **terminate** action for a service enables you to more tightly control which PPPoE clients can access and receive services from a particular PPPoE server.
  - **delay**—Number of seconds that the PPPoE underlying interface waits after receiving a PADI packet from the client before sending a PADO packet in response. In networks with mesh topologies, you might want to designate a primary PPPoE server and a backup PPPoE server for handling a particular service request. In such a scenario, you can configure a delay for the associated service entry on the backup PPPoE server to allow sufficient time for the primary PPPoE server to respond to the client with a PADO packet. If the primary server does not send the PADO packet within the delay period configured on the backup server, then the backup server sends the PADO packet after the delay period expires.
  - **drop**—Directs the router to drop (ignore) a PADI packet containing the specified service name tag when received from a PPPoE client, which effectively denies the client's request to provide the associated service. The **any** service is associated with the **drop** action by default. To prohibit the router from responding to PADI packets that contain **empty** or **any** service name tags, you can configure the **drop** action for the empty or **any** service. You can also use the **drop** action in combination with ACI/ARI pairs to accept specific service name tags only from specific subscribers, as described in the following information about ACI/ARI pairs.

ACI/ARI Pairs in PPPoE Service Name Tables

To specify agent circuit identifier (ACI) and agent remote identifier (ARI) information for a named service, **empty** service, or **any** service in a PPPoE service name table, you can configure an ACI/ARI pair. An ACI/ARI pair contains an agent circuit ID string that identifies the DSLAM interface that initiated the service request, and an agent remote ID string that identifies the subscriber on the DSLAM interface that initiated the service request. You can think of an ACI/ARI pair as the representation of one or more PPPoE clients accessing the router by means of the PPPoE service name table.

ACI/ARI specifications support the use of wildcard characters in certain formats. You can configure a combined maximum of 8000 ACI/ARI pairs, both with and without wildcards, per PPPoE service name table. You can distribute the ACI/ARI pairs in any combination among the service entries in the service name table.

You must specify the action—**terminate**, **delay**, or **drop**—taken by the underlying PPPoE interface when it receives a client request containing vendor-specific ACI/ARI information that matches the ACI/ARI information configured in the PPPoE service name table on the router. An ACI/ARI pair is associated with the **terminate** action by default.

For example, assume that for the **user1-service** named service, you configure the **drop** action for the service and the **terminate** action for the associated ACI/ARI pairs. In this case, the ACI/ARI pairs identify the DSLAM interfaces and associated subscribers authorized to access the PPPoE server. Using this configuration causes the router to drop PADI packets containing the **user1-service** tag *unless* the PADI packet also contains vendor-specific ACI/ARI information that matches the subscribers identified in one or more of the ACI/ARI pairs. For PADI packets containing matching ACI/ARI information, the router sends an immediate PADO response to the client indicating that it can provide the requested service for the specified subscribers.

You can also associate a PPPoE dynamic profile, routing instance, and static PPPoE interface with an ACI/ARI pair.

Dynamic Profiles and Routing Instances in PPPoE Service Name Tables

You can associate a previously configured PPPoE dynamic profile with a named service, **empty** service, or **any** service in the PPPoE service name table, or with an ACI/ARI pair defined for these services. The router uses the attributes defined in the profile to instantiate a dynamic PPPoE subscriber interface based on the service name, ACI, and ARI information provided by the PPPoE client during PPPoE negotiation. The dynamic profile configured for a service entry or ACI/ARI pair in a PPPoE service name table overrides the dynamic profile assigned to the PPPoE underlying interface on which the dynamic PPPoE interface is created.

To specify the routing instance in which to instantiate the dynamic PPPoE interface, you can associate a previously configured routing instance with a named service, **empty** service, or **any** service in the PPPoE service name table, or with an ACI/ARI pair defined for these services. Like dynamic profiles configured for service entries or ACI/ARI pairs, the routing instance configured for the PPPoE service name table overrides the routing instance assigned to the PPPoE underlying interface.

For information about configuring the PPPoE service name table to create a dynamic PPPoE subscriber interface, see "Assigning a Dynamic Profile and Routing Instance to a Service Name or ACI/ARI Pair for Dynamic PPPoE Interface Creation" on page 150.

Maximum Sessions Limit in PPPoE Service Name Tables

To limit the number of PPPoE client sessions that can use a particular service entry in the PPPoE service name table, you can configure the maximum number of active PPPoE sessions using either dynamically-created or statically-created PPPoE interfaces that the router can establish with a particular named service, **empty** service, or **any** service.

(You cannot configure the maximum sessions limit for an ACI/ARI pair.) The maximum sessions limit must be in the range 1 through the platform-specific maximum PPPoE sessions supported for your routing platform. The router maintains a count of active PPPoE sessions for each service entry to determine when the maximum sessions limit has been reached.

The router uses the maximum sessions value for a service entry in the PPPoE service name table in conjunction with both of the following:

- The maximum sessions (**max-sessions**) value configured for the PPPoE underlying interface
- The maximum number of PPPoE sessions supported on your routing platform

If your configuration exceeds either of these maximum session limits, the router cannot establish the PPPoE session.

Static PPPoE Interfaces in PPPoE Service Name Tables

To reserve a previously configured static PPPoE interface for use only by the PPPoE client with matching ACI/ARI information, you can specify a single static PPPoE interface for each ACI/ARI pair defined for a named service entry, **empty** service entry, or **any** service entry in a PPPoE service name table. (You cannot configure a static interface for a service entry that does not have an ACI/ARI pair defined.) The static PPPoE interface associated with an ACI/ARI pair takes precedence over the general pool of static PPPoE interfaces associated with the PPPoE underlying interface configured on the router.

When you configure a static interface in the PPPoE service name table, make sure there is a one-to-one correspondence between the PPPoE client and the static interface. For example, if two clients have identical ACI/ARI information that matches the information in the PPPoE service name table, the router reserves the static interface for exclusive use by the first client that logs in to the router. As a result, the router prevents the second client from logging in.

NOTE: **You cannot configure a static interface for an ACI/ARI pair already configured with a dynamic profile and routing instance. Conversely, you cannot configure a dynamic profile and routing instance for an ACI/ARI pair already configured with a static interface.**

PADO Advertisement of Named Services in PPPoE Service Name Tables

By default, the advertisement of named services in PADO control packets sent by the router to the PPPoE client is disabled. You can enable advertisement of named services in the PADO packet as a global option when you configure the PPPoE protocol on the router. Configuring PADO advertisement notifies PPPoE clients of the services that the router (AC) can offer.

If you enable advertisement of named services in PADO packets, make sure the number and length of all advertised service entries does not exceed the maximum transmission unit (MTU) size supported by the PPPoE underlying interface.

Related Documentation

- Evaluation Order for Matching Client Information in PPPoE Service Name Tables on page 206
- *Benefits of Configuring PPPoE Service Name Tables*
- *Configuring PPPoE Service Name Tables*
- *Example: Configuring a PPPoE Service Name Table*
- Configuring Dynamic PPPoE Subscriber Interfaces Using Dynamic Profiles on page 145
- *PPPoE Overview*

Evaluation Order for Matching Client Information in PPPoE Service Name Tables

When the router receives a service request from a PPPoE client, it evaluates the entries configured in the PPPoE service name table to find a match for the client's ACI/ARI information so it can take the appropriate action.

The order of evaluation is as follows:

1. The router evaluates the ACI/ARI information configured for the **any** service entry, and ignores the contents of the service name tag transmitted by the client.
2. If no match is found for the client information, the router evaluates the ACI/ARI information for the **empty** service entry and the named service entries. If an ACI/ARI pair is not configured for these service entries, the router evaluates the other attributes configured for the **empty** service and named services.
3. If there is still no match for the client information, the router evaluates the other attributes configured for the **any** service entry, and ignores both the ACI/ARI information for the **any** service and the contents of the service name tag transmitted by the client. If the **any** service is configured for the default action, **drop**, the router drops the PADR packet. If the **any** service is configured for a nondefault action (**terminate** or **delay**), the router evaluates the other attributes configured for the **any** service.

Related Documentation

- Understanding PPPoE Service Name Tables on page 201
- *Benefits of Configuring PPPoE Service Name Tables*
- *Configuring PPPoE Service Name Tables*
- Example: Configuring a PPPoE Service Name Table for Dynamic Subscriber Interface Creation on page 207
- *PPPoE Overview*
- Dynamic PPPoE Subscriber Interfaces over Static Underlying Interfaces Overview on page 153

Example: Configuring a PPPoE Service Name Table for Dynamic Subscriber Interface Creation

This example shows how to configure a PPPoE service name table to create a dynamic PPPoE subscriber interface based on the service name, agent circuit identifier (ACI), and agent remote identifier (ARI) information provided by PPPoE clients during PPPoE negotiation.

In this example, PPPoE service name table **TableDynamicPPPoE** includes an **any** service entry, **empty** service entry, and two named service entries: **Premium** and **Standard**. The PPPoE underlying interfaces configured for **TableDynamicPPPoE** are **ge-2/0/0.1** and **ge-2/0/0.2**. Only **ge-2/0/0.1** is configured for dynamic profile assignment and creation of dynamic PPPoE subscriber interfaces.

Following the configuration example, Table 7 on page 209 explains how the router evaluates the entries in **TableDynamicPPPoE** to create a dynamic PPPoE subscriber interface in a specified routing instance for each of several sample clients.

To configure a PPPoE service name table to create dynamic PPPoE subscriber interfaces:

1. Configure the PPPoE service name table.

```
protocols {
  pppoe {
    service-name-tables TableDynamicPPPoE {
      service any {
        terminate;
        max-sessions 100;
        dynamic-profile AnyProfile;
        agent-specifier {
          aci "broadway-ge-1/0/1.0" ari "london" {
            terminate;
            dynamic-profile LondonProfile;
            routing-instance LondonRI;
          }
          aci "groton-ge-4/0/3.32" ari "paris" {
            delay 5;
            dynamic-profile ParisProfile;
            routing-instance ParisRI;
          }
        }
      }
      service empty {
        drop;
        agent-specifier {
          aci "dunstable-ge-1/0/0.1" ari "kanata" {
            dynamic-profile BasicPppoeProfile;
            delay 10;
          }
        }
      }
      service Premium {
        terminate;
```



```
          dynamic-profile PremiumProfile;
        }
        service Standard {
          terminate;
          max-sessions 10;
          dynamic-profile StandardProfile;
          agent-specifier {
            aci "dunstable-ge-1/0/0.1" ari "kanata" {
              dynamic-profile BasicPppoeProfile;
              delay 10;
            }
          }
        }
      }
    }
  }
```

2. Configure the PPPoE underlying interface for the service name table.

```
interfaces {
  ge-2/0/0 {
    vlan-tagging;
    unit 1 {
      vlan-id 1;
      pppoe-underlying-options {
        dynamic-profile BasicPppoeProfile;
        service-name-table TableDynamicPPPoE;
      }
    }
    unit 2 {
      vlan-id 2;
      pppoe-underlying-options {
        service-name-table TableDynamicPPPoE;
      }
    }
  }
}
```

Table 7 on page 209 lists the service name, ACI value, and ARI value provided in several sample PPPoE client requests, and the name of the PPPoE underlying interface on which the router received each client request. The Results column describes the dynamic PPPoE subscriber interface created by the router based on *both* of the following:

- The values received from each PPPoE client during PPPoE negotiation
- The sequence in which the router evaluates the entries configured in the PPPoE service name table to find a match for the client's service name and ACI/ARI information, as described in "Evaluation Order for Matching Client Information in PPPoE Service Name Tables" on page 206

Table 7: Dynamic PPPoE Subscriber Interface Creation Based on PPPoE Client Request Values

PPPoE Client	Service Name	ACI Value	ARI Value	Receiving Underlying Interface	Results
Client 1	Premium	broadway-ge-1/0/1.1	london	ge-2/0/0.1	Matches ACI/ARI pair configured for **any** service. Router creates dynamic PPPoE subscriber interface over **ge-2/0/0.1** using **LondonProfile** dynamic profile and **LondonRI** routing instance assigned to **any** service.
Client 2	Premium	dunstable-ge-1/0/1.0	toronto	ge-2/0/0.1	Matches base **Premium** service. Router creates dynamic PPPoE subscriber interface over **ge-2/0/0.1** using **PremiumProfile** dynamic profile and routing instance associated with **ge-2/0/0.1** underlying interface.
Client 3	empty	dunstable-ge-1/0/0.1	kanata	ge-2/0/0.1	Matches ACI/ARI pair configured for **empty** service and **Standard** service. Router creates dynamic PPPoE subscriber interface over **ge-2/0/0.1** after a delay of 10 seconds. Router uses **BasicPPPoEProfile** dynamic profile and routing instance associated with **ge-2/0/0.1** underlying interface.
Client 4	empty	slinger-ge-1/0/0.1	chicago	ge-2/0/0.2	Because receiving underlying interface **ge-2/0/0.2** is *not* associated with a dynamic profile, router does not create a dynamic PPPoE subscriber interface, and drops any PADI or PADR control packets received from this client.
Client 5	Standard	slinger-ge-1/0/0.1	chicago	ge-2/0/0.1	Matches base **Standard** service. Router creates dynamic PPPoE subscriber interface over **ge-2/0/0.1** using **StandardProfile** dynamic profile and routing instance associated with **ge-2/0/0.1** underlying interface.

Related Documentation

- Evaluation Order for Matching Client Information in PPPoE Service Name Tables on page 206
- Subscriber Interfaces and PPPoE Overview on page 141
- Understanding PPPoE Service Name Tables on page 201

- *Configuring PPPoE Service Name Tables*

CHAPTER 21

Monitoring and Managing Dynamic PPPoE for Subscriber Access

Verifying and Managing Dynamic PPPoE Configuration

Purpose View or clear information about dynamic PPPoE logical interfaces, underlying interfaces for dynamic PPPoE logical interfaces, and PPPoE statistics.

Action

- To display information about the properties of all PPPoE underlying interfaces associated with a dynamic PPPoE profile:

  user@host> **show pppoe underlying-interfaces**

- To display information about the PPPoE properties of a specified underlying interface associated with a dynamic PPPoE profile:

  user@host> **show pppoe underlying-interfaces *interface-name***

- To display session-specific information about PPPoE interfaces, including whether the interface was dynamically created or statically created:

  user@host> **show pppoe interfaces**

- To display information for a specified PPPoE service name table, including the assigned dynamic profile and routing instance, if configured:

  user@ host> **show pppoe service-name-tables *table-name***

- To display information about all active PPPoE sessions on the router:

  user@host > **show pppoe sessions**

- To display information for all active PPPoE sessions established for a specified service name:

  user@host > **show pppoe sessions service *service-name***

- To display information for all active PPPoE sessions established for a specified agent circuit identifier (ACI) or agent remote identifier (ARI) string:

  user@host > **show pppoe sessions aci "west-ge-2/0/3"**
  user@host > **show pppoe sessions ari "sunnyvale"**

- To display PPPoE control packet statistics for all PPPoE sessions:

user@host> **show pppoe statistics**

- To display PPPoE control packet statistics for a specified PPPoE underlying interface:

  user@host> **show pppoe statistics** ***interface-name***

- To clear (reset) PPPoE control packet statistics for all PPPoE sessions:

  user@host> **clear pppoe statistics**

- To clear (reset) PPPoE control packet statistics for a specified underlying Ethernet interface:

  user@host> **clear pppoe statistics** ***underlying-interface-name***

- To display summary information about PPPoE subscriber sessions currently undergoing lockout or currently in a lockout grace period on all PPPoE underlying interfaces:

  user@host> **show pppoe lockout**

- To display summary information about PPPoE subscriber sessions currently undergoing lockout or currently in a lockout grace period on the specified PPPoE underlying interface:

  user@host> **show pppoe lockout** ***underlying-interface-name***

Related Documentation

- CLI Explorer

PART 4

Configuring MLPPP for Subscriber Access

CHAPTER 22

MLPPP Support for LNS and PPPoE Subscribers Overview

MLPPP Overview

Multilink Point-to-Point Protocol (MLPPP) aggregates multiple PPP physical links into a single virtual connection, or logical bundle. More specifically, MLPPP bundles multiple link-layer channels into a single network-layer channel. Peers negotiate MLPPP during the initial phase of Link Control Protocol (LCP) option negotiation. Each router indicates that it is multilink capable by sending the multilink option as part of its initial LCP configuration request.

An MLPPP bundle can consist of multiple physical links of the same type—such as multiple asynchronous lines—or can consist of physical links of different types—such as leased synchronous lines and dial-up asynchronous lines.

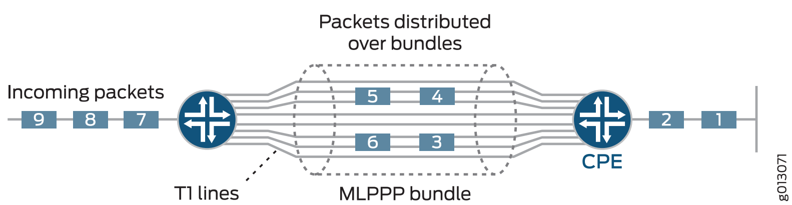
Packets received with an MLPPP header are subject to fragmentation, reassembly, and sequencing. Packets received without the MLPPP header cannot be sequenced and can be delivered only on a first-come, first-served basis.

This section contains the following topics:

Traditional MLPPP Application

MLPPP is used to bundle multiple low speed links to create a higher bandwidth pipe such that the combined bandwidth is available to traffics from all links, and to support link fragmentation and interleaving (LFI) support on the bundle to reduce the transmission delay of high priority packets. LFI interleaves voice packets with fragmented data packets to ensure timely delivery of voice packets. Figure 4 on page 216 shows how incoming packets are distributed and aggregated into an MLPPP bundle.

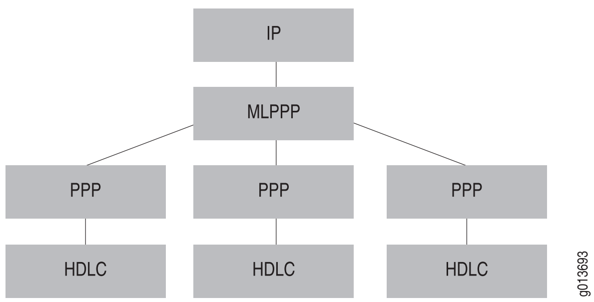
Figure 4: MLPPP Aggregation of Traffic Into Single Bundle



Because MLPPP aggregates multiple link-layer channels onto a single network-layer IP interface, protocol layering within the router is different than for non-multilink PPP.

Figure 5 on page 216 illustrates interface stacking with MLPPP.

Figure 5: Structure of MLPPP



MLPPP LCP Negotiation Option

Multilink PPP adds the multilink maximum received reconstructed unit (MRRU) option for LCP negotiation. The MRRU option has two functions:

- It informs the other end of the link the maximum reassembled size of the PPP packet payload that the router can receive.
- It informs the other end that the router supports MLPPP.

When you enable multilink on your router, the router includes the MRRU option in LCP negotiation with the default value set to 1500 bytes (user-configurable option) for PPP. If the remote system rejects this option, the local system determines that the remote system does not support multilink PPP and it terminates the link without negotiation.

NOTE: The router does not bring up a link if the MRU value received from a peer device differs from the MRRU value received from the peer.

Related Documentation

- MLPPP Support for LNS and PPPoE Subscribers on MX Series Overview on page 217
- Supported Features for MLPPP LNS and PPPoE Subscribers on the MX Series on page 219
- Understanding MLPPP Link Fragmentation and Interleaving on page 223

MLPPP Support for LNS and PPPoE Subscribers on MX Series Overview

Multilink PPP (MLPPP) support is provided to LNS (L2TP network server) and PPPoE (Point-to-Point Protocol over Ethernet) terminated and tunneled subscribers running on MX Series with access-facing MPC2s.

For customers with both MLPPP and single link PPP clients, the MX Series needs to determine client capability during link control protocol (LCP) negotiation and support either multilink or single link access modules accordingly (mixed mode support).

This section contains the following topics:

Single Member Link MLPPP Bundle Support

MLPPP running on the MX Series provides link fragmentation and interleaving (LFI) support for a single-link bundle. Each bundle contains a single member link only; configuration of multiple member links belonging to the same bundle are rejected. However, LFI enables the single subscriber session to send small, high priority packets interleaving with large packets without introducing unacceptable transmission delay for high priority small packets. LFI interleaves voice packets with fragmented data packets to ensure timely delivery of voice packets and to guarantee voice quality.

Customers with lower bandwidth subscribers benefit from the MLPPP LFI support. With the traditional non-MLPPP application, the CPE (customer premises equipment) device performs the fragmentation prior to the PPP encapsulation and then relies on the application at the far end to perform the reassembly. With the MLPPP solution, the burden to reassemble the packets on the customer servers and the far-end application is removed, and control is given to the service provider for fragmentation and reassembly.

NOTE: A maximum of 8000 MLPPP bundles is supported.

Member Link and Bundle Configuration

An MLPPP subscriber consists of two IFLs (logical interfaces), a member link, and a bundle. For MLPPP subscribers, you can configure the member link and bundle statically, or dynamically using dynamic profiles.

- Static MLPPP Subscribers—You must configure both member link and bundle IFLs manually before the member link IFL can start LCP (link control protocol) negotiation either for an LNS session or for a PPPoE session.
- Dynamic MLPPP Subscribers—You configure dynamic member IFLs using dynamic profiles. The member link dynamic profile includes the **family mlppp** statement

containing the bundle dynamic profile and the service interface (**si**), or a pool of service interfaces. This information is then used to create the dynamic bundle IFL.

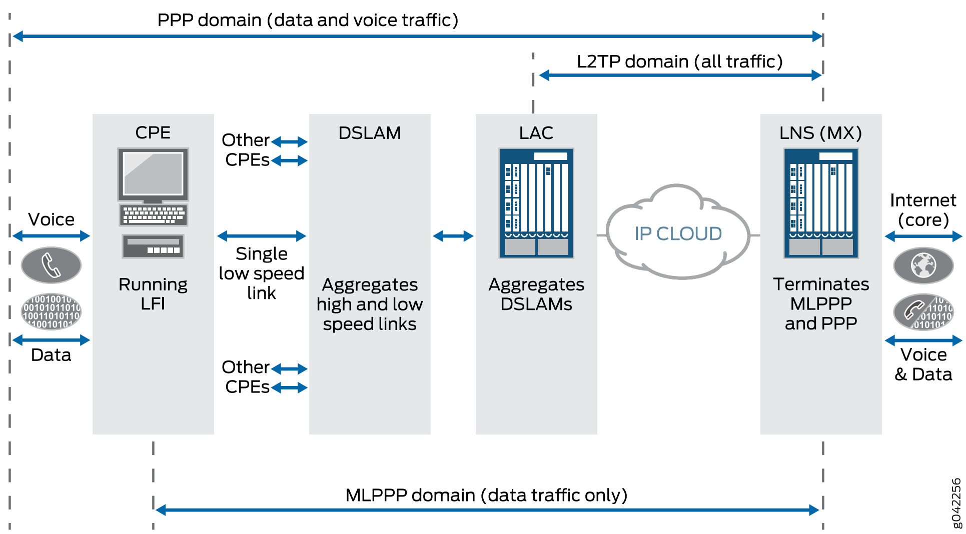
Each bundle accepts only one member link. If more than one member link attempts to join the same bundle, the system fails the new member session.

Dual-stack is supported for the bundle.

LNS Subscribers and MX Series

Figure 6 on page 218 shows a network diagram with the MX Series functioning as the LNS. Both PPP and MLPPP bundles are terminated at the LNS.

Figure 6: MLPPP Bundles Terminated at MX Series as the LNS Network



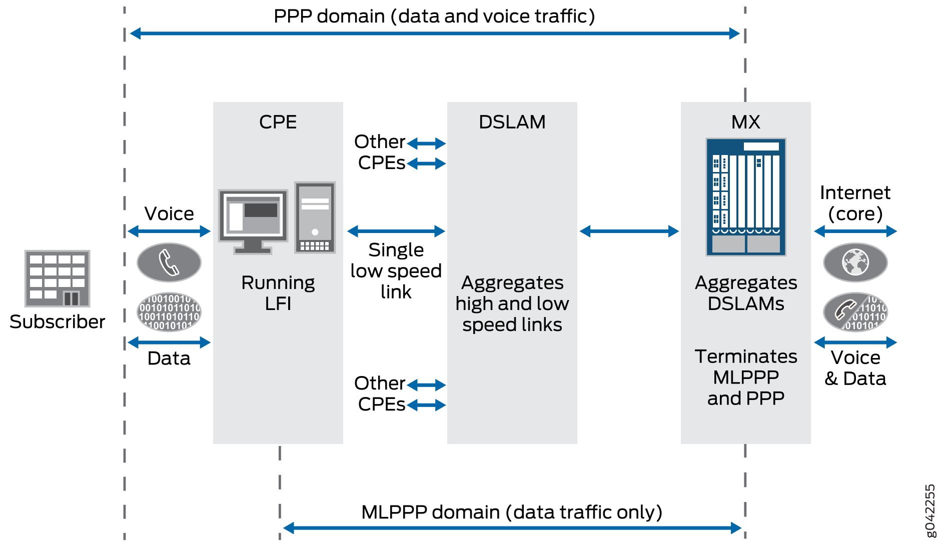
The following three domains are shown passing traffic through the LNS network:

- PPP domain—Contains data and voice traffic
- MLPPP domain—Contains data traffic only
- L2TP domain—Contains all types of traffic

PPPoE Subscribers and MX Series

Figure 7 on page 219 shows a network diagram with the MX Series terminating PPPoE sessions that include both the PPP and MLPPP bundles.

Figure 7: PPPoE Sessions Terminated at MX Series



The following two domains are shown passing traffic through the network:

- PPP domain—Contains data and voice traffic
- MLPPP domain—Contains data traffic only

Related Documentation

- MLPPP Overview on page 215
- Supported Features for MLPPP LNS and PPPoE Subscribers on the MX Series on page 219
- Mixed Mode Support for MLPPP and PPP Subscribers Overview on page 220
- MLPPP Bundles and Inline Service Logical Interfaces Overview on page 233

Supported Features for MLPPP LNS and PPPoE Subscribers on the MX Series

The following features are supported for multilink PPP (MLPPP) for L2TP network server (LNS) and Point-to-Point Protocol over Ethernet (PPPoE, terminated and tunneled) subscribers on the MX Series:

- Supports MLPPP for static and dynamic LNS subscribers and PPPoE subscribers.
- Supports each MLPPP bundle containing a single member link.
- Anchors the bundle logical interface (IFL) on the inline services **si** interface.
- Runs the bundle IFL on an MX Series that enables shaping and queuing at the bundle to minimize fragment reordering.
- Supports configurable service device pools for load-balancing bundle IFLs.

- Supports the co-existence for member link IFL and the bundle IFL on different lookup engines.
- Supports fragmentation maps for both static and dynamic **si** interfaces, and supports multiple forwarding classes pointing to a single queue for **si** interface attachments.
- Provides fragmentation of low-priority packets towards the subscriber, and reassembly of low-priority packets towards the core, and availability of per-bundle fragmentation and reassembly statistics.
- Supports bundle **family inet** and **family inet6**, including DHCPv6 prefix delegation over MLPPP bundle for both LNS and PPPoE MLPPP subscribers.
- Supports lawful intercept over MLPPP bundles.
- Provides mixed mode (PPP and MLPPP) support for subscribers.
- Maintains existing LNS and PPPoE subscriber management functionalities.
- Supports graceful Routing Engine switchover (GRES).

Related Documentation

- MLPPP Support for LNS and PPPoE Subscribers on MX Series Overview on page 217
- Mixed Mode Support for MLPPP and PPP Subscribers Overview on page 220
- MLPPP Bundles and Inline Service Logical Interfaces Overview on page 233

Mixed Mode Support for MLPPP and PPP Subscribers Overview

Existing customer edge subscriber services separate MLPPP and PPP support for subscribers. However, if a subscriber interface is configured for MLPPP and the customer premises equipment (CPE) does not support MLPPP, then the subscriber login fails.

In an environment where MLPPP and PPP subscribers are mixed and you cannot easily manage the subscriber types by classifying them into separate groups using dynamic profiles, the MX Series needs the capability to renegotiate Link Control Protocol (LCP) in PPP if the CPE rejects LCP negotiation in MLPPP. This capability is known as *mixed mode support*.

Mixed mode uses common configuration and flexibility to support PPP and MLPPP. If you configure a subscriber interface using the **family mlppp** and **family inet/inet6** statements for PPP-only CPE, mixed mode support enables additional LCP negotiation exchanges to successfully negotiate LCP in PPP. Mixed mode supports static and dynamic PPPoE (terminated and tunneled) and LNS (L2TP network server) subscribers.

This section contains the following topics:

- PPPoE Terminated and Tunneled Subscribers on page 221
- LNS Subscribers on page 221

PPPoE Terminated and Tunneled Subscribers

If you do not configure the **family mlppp** statement for a subscriber interface, the MX Series negotiates LCP in PPP as it currently does, and any LCP request that contains MLPPP options is rejected.

However for PPPoE subscribers, if you configure the **family mlppp** statement for a subscriber interface, the MX Series negotiates LCP in MLPPP with the CPE. If the CPE rejects MLPPP, then the MX Series renegotiates LCP in PPP with the CPE.

Mixed mode operation for a LAC (tunneled PPPoE) subscriber is the same as for a terminated PPPoE subscriber. The authentication phase has no effect on LAC mixed mode operation because LCP negotiation must be completed prior to authentication.

LNS Subscribers

For LNS subscribers, the MX Series negotiates LCP as follows:

- If proxy data from the LAC indicates that MLPPP was negotiated, and the proxy data is acceptable, and the **lcp-renegotiation** statement is not configured, then the proxy is accepted and the subscriber is MLPPP.
- If proxy data from the LAC indicates that PPP was negotiated, or if there was no proxy data from LAC, or if the **lcp-renegotiation** statement is configured for the LAC, then the MX Series starts LCP negotiation in MLPPP with the CPE.

  If the CPE rejects MLPPP, then the MX Series renegotiates LCP in PPP with the CPE.

Related Documentation

- MLPPP Support for LNS and PPPoE Subscribers on MX Series Overview on page 217
- Supported Features for MLPPP LNS and PPPoE Subscribers on the MX Series on page 219
- Configuring L2TP Client Access to Support MLPPP for Static Subscribers on page 239
- Example: Configuring Dynamic LNS MLPPP Subscribers on page 269

CHAPTER 23

Configuring MLPPP Link Fragmentation and Interleaving

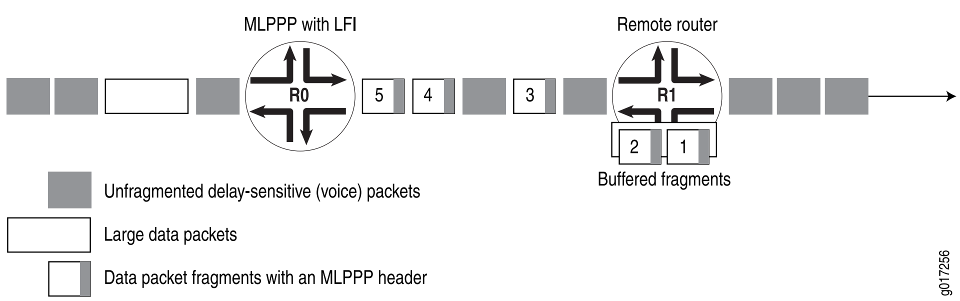
Understanding MLPPP Link Fragmentation and Interleaving

Priority scheduling on a multilink (MLPPP) 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 MLPPP 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 by reducing 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 MLPPP bundle.

Figure 8 on page 223 shows how LFI processes packets.

Figure 8: LFI Packet Processing



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.
- The remaining non-LFI (data) packets are fragmented or unfragmented based on the configured fragmentation threshold. 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.

Fragmentation is performed according to the fragmentation threshold that you configure. For example, if you configure a fragmentation threshold of 128 bytes, all packets greater 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. Device R1 transmits the unfragmented data packets as it receives them and does not buffer them.

Related Documentation

- Understanding MLPPP and Fragmentation-Maps on page 224
- Understanding Fragmented Packet Queuing on page 227
- Understanding Sequenced Packet Fragment Drops on page 231

Understanding MLPPP and Fragmentation-Maps

You enable link fragmentation and interleaving (LFI) on inline service (**si**) interface bundles by configuring **fragmentation-maps**. For multilink PPP (MLPPP) bundle support, you must configure **fragmentation-maps** in **class-of-services** and reference them in either the bundle dynamic-profile or bundle logical interface (IFL) configuration.

BEST PRACTICE: For MX Series and class-of-service (CoS) implementation, you can configure a fragmentation map to have two forwarding classes pointing to the same queue. However, if you assign multiple forwarding classes to a single queue, you must also reference all of those forwarding classes in a fragmentation map to enable the expected behavior.

If you reference only one of the forwarding classes assigned to a queue, then the other forwarding classes in that queue can clog that queue with large packets. For previous existing fragmentation-map implementations, this condition did not occur because the other forwarding classes inherited this fragmentation behavior assigned to that queue.

If you assign multiple forwarding classes to a queue, create a fragmentation map that addresses each of those forwarding classes. This results in fragmentation-map behavior that more closely reflects the expected behavior based on the fragmentation CLI, while the existing fragmentation-map behavior remains unchanged.

This section contains the following topics:

Fragmentation-Map Settings

By setting **fragmentation-maps** under **class-of-service**, you can configure the fragmentation properties on a particular forwarding class, as shown in the following sample output:

```
class-of-service {
 fragmentation-maps {
  map-name {
   forwarding-class class-name {
    fragment-threshold bytes;
    no-fragmentation;
   }
  }
 }
}
```

NOTE: The per-forwarding class **drop-timeout** statement enabling you to change the resequencing interval in milliseconds for each fragmentation class is not supported in the fragmentation map.

You can configure the following settings for **fragmentation-maps**:

- (Optional) **fragment-threshold**—Sets a per-forwarding class fragmentation threshold in bytes. **fragment-threshold** sets the maximum size of each multilink fragment. An extra MLPPP header is prepended to these multilink fragments. This same header is also prepended to packets of these forwarding classes that are smaller than the fragmentation threshold.
  - For MLPPP bundle interface configuration, you can set the **fragment-threshold** for all forwarding classes. Any fragmentation threshold defined by a **fragmentation-map** and applied to that interface takes precedence for the forwarding classes referenced by that **fragmentation-map**.
  - For **si** bundle IFL configuration, the **fragment-threshold** applies to all forwarding classes. The **fragment-threshold** setting in **fragmentation-maps** for a particular forwarding class, if configured, overrides the threshold configured in **si** bundle IFL for that class. If no **fragment-threshold** is configured anywhere, packets are still fragmented if the threshold exceeds the smallest MTU or MRRU of all links in the bundle.

    NOTE: The per-forwarding class **multilink-class** statement enabling you to map a forwarding class into a multiclass MLPPP is not supported for **si** MLPPP bundles.

- (Required) **no-fragmentation**—Sets traffic on a particular forwarding class to be interleaved rather than fragmented. The **no-fragmentation** setting is required to define high priority traffic and indicates that an extra fragmentation header is not prepended to the packets of this forwarding class

NOTE: For a given forwarding class, you can include either the **fragment-threshold** setting or the **no-fragmentation** setting; they are mutually exclusive.

Understanding Fragmentation-Map Bindings

Using MLPPP in this manner generates two subscriber interfaces for each subscriber:

- The inline services (**si**) bundle interface IFL.
- The PPP member link IFL.

The data plane traffic destined for the subscriber exits through the (**si**) bundle interface IFL, and passes through the PPP member link IFL. Queuing is provided for both of these IFLs, which then requires the ability to define class of service.

When you are creating the two subscriber interfaces, the MX Series authenticates only a single user, and the RADIUS server only provides a single set of class-of-service (CoS) attributes. These CoS RADIUS attributes are then applied to both the (**si**) bundle interface IFL and the PPP member link IFL.

NOTE: For this scenario to succeed, you must have already configured the dynamic profiles for these IFLs to accept CoS RADIUS attributes enabling both the (si) bundle interface IFL and the PPP member link IFL to have the same CoS attributes.

To apply different CoS to the (**si**) bundle interface IFL and the PPP member link IFL, you can set CoS RADIUS attributes to specify the Transmission Control Protocol (TCP) name to which the attribute is intended. The dynamic profile associated with the (**si**) bundle interface IFL contains the CoS TCP for that IFL, and the dynamic profile associated with the PPP member link IFL contains the CoS TCP for that IFL.

The RADIUS attributes each include a target TCP. When configured, two sets of CoS RADIUS attributes are retrieved with the member link authentication; one set with the (**si**) bundle interface IFL TCP specified, and the other set with the PPP member link IFL TCP specified.

Related Documentation

- *fragmentation-maps*
- Understanding MLPPP Link Fragmentation and Interleaving on page 223
- Understanding Fragmented Packet Queuing on page 227
- Understanding Sequenced Packet Fragment Drops on page 231

Understanding Fragmented Packet Queuing

Fragmented Multilink PPP (MLPPP) packets have a multilink header containing a multilink sequence number. The sequence numbers on these fragments must be preserved so that the remote device receiving these fragments can correctly reassemble them into a complete packet. To accommodate this requirement, Junos OS queues all packets on member links of a multilink bundle with a MLPPP header into a single queue (q0) by default.

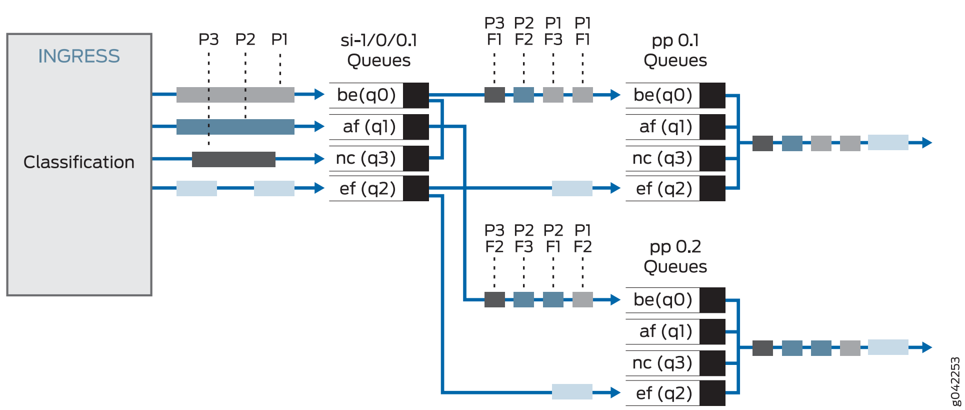
- Traffic flows of a forwarding class that has MLPPP fragmentation configured are distributed from the inline services **si** bundle interface queues to the member link queues (queue 0) following a round-robin method.
- Traffic flows of a forwarding class without MLPPP fragmentation are distributed from the **si** bundle interface queues to the member link queues based on a hashing algorithm computed from the destination address, source address, and IP protocol of the packet.

  If the IP payload contains TCP or UDP traffic, the hashing algorithm also includes the source and destination ports. As a result, all traffic belonging to one traffic flow is queued to one member link.

Figure 9 on page 228 shows how traffic is queued on an MLPPP multilink bundle and its member links. Packet flows in the figure use the notation P*x*,F*x*; for example, P1,F1 represents Packet 1, Fragment 1.

- There are four queues.
- Forwarding classes be, af, and nc are mapped to queues q0, q1, and q3, respectively, on the multilink bundle. These are fragmented.
- Forwarding class ef contains voice traffic, and is mapped to q2 and is not fragmented.
- Interface **si-1/0/0.1** is the bundle, and **pp0.1** and **pp0.2** are the member links for that bundle.

Figure 9: Queuing on Member Links



Queuing on member links proceeds as follows:

1. The packet fragments of forwarding classes be, af, and nc on the multilink bundle are mapped to q0 on Member Links 1 and 2. These packets are distributed from the **si** queues to the member links using a round-robin method.
2. The packets of forwarding class ef (voice) from the multilink bundle are mapped to q2 on the member links. This forwarding class is not fragmented. The packets are distributed from the **si** queues to the member links based on a hashing algorithm.
3. The network control packets from the multilink bundle are mapped to q0 on the member links. The bundle network control traffic is queued with the data flows on the member link. However, q3 on the member links transmits network control packets that exchange protocol information related to member links, such as packets exchanging hello messages on member links.

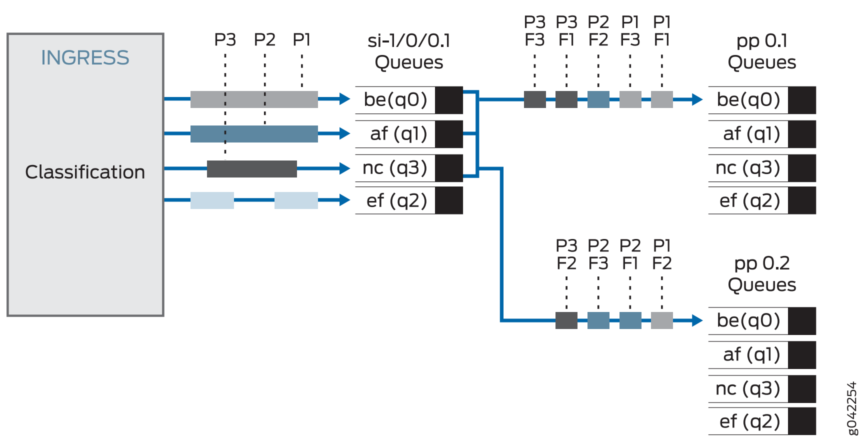
This section contains the following topics:

Queuing of Fragmented Packets to Member Links

On a multilink bundle, packet fragments from all forwarding classes with fragmentation enabled are transmitted to q0 on member links. On the q0 queues of member links, packets are queued using a round-robin method to enable per-fragment load balancing.

Figure 10 on page 229 shows how fragmented packet queuing is performed on the member links. Packet flows in the figure use the notation P*x*,F*x*; for example, P1,F1 represents Packet 1, Fragment 1.

Figure 10: Queuing of Fragmented Packets on Member Links



Packet fragments from the multilink bundle are queued to member links one by one using a round-robin method:

- Packet P1,F1 from q0 on the multilink bundle is queued to q0 on Member Link 1.
- Packet P1,F2 from q0 on the multilink bundle is queued to q0 on Member Link 2.

- Packet P1,F3 from q0 on the multilink bundle is queued to q0 on Member Link 1.
- Packet P2,F1 from q1 on the multilink bundle is queued to q0 on Member Link 2, and so on.

NOTE: **Packets that are part of the fragmented forwarding class, but are not fragmented, follow the same procedure.**

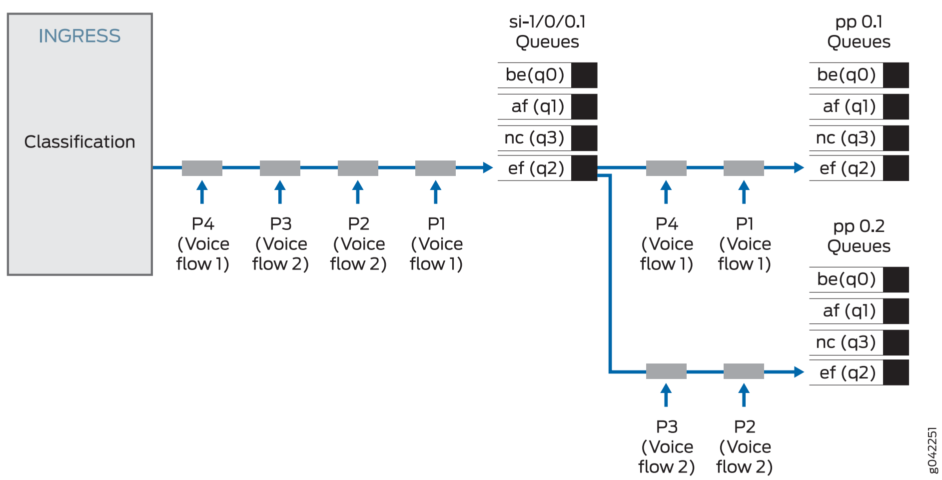
After exiting the **si** interface, Microcode adds a header of approximately 40 bytes to the MLPPP packets. When configuring the class-of-service shaping, you may need to adjust bytes to account for this.

Queuing of LFI Packets to Member Links

On a multilink bundle, all non-MLPPP encapsulated traffic [link fragmenting and interleaving (LFI) traffic] from the multilink bundle are queued to the queue as defined by the forwarding class of that packet.

Figure 11 on page 230 shows how LFI packet queuing is performed on the member links.

Figure 11: Queuing of LFI Packets on Member Links



The packets are distributed from the **si** interface to the member links based on a hashing algorithm computed from the source address, destination address, and IP protocol of the packet.

If the IP payload contains TCP or UDP traffic, the hashing algorithm also includes the source and destination ports. As a result, all traffic belonging to one traffic flow is queued to one member link.

Related Documentation

- Understanding MLPPP Link Fragmentation and Interleaving on page 223
- Understanding MLPPP and Fragmentation-Maps on page 224
- Understanding Sequenced Packet Fragment Drops on page 231

Understanding Sequenced Packet Fragment Drops

Multilink PPP (MLPPP) link fragmentation and interleaving (LFI) provides buffering at the receiver side of a link to reassemble MLPPP fragmented packets. Dropping of the packet fragments is a concern because the packet fragments' remainder consumes valuable bandwidth and buffer space, only to have it eventually being dropped.

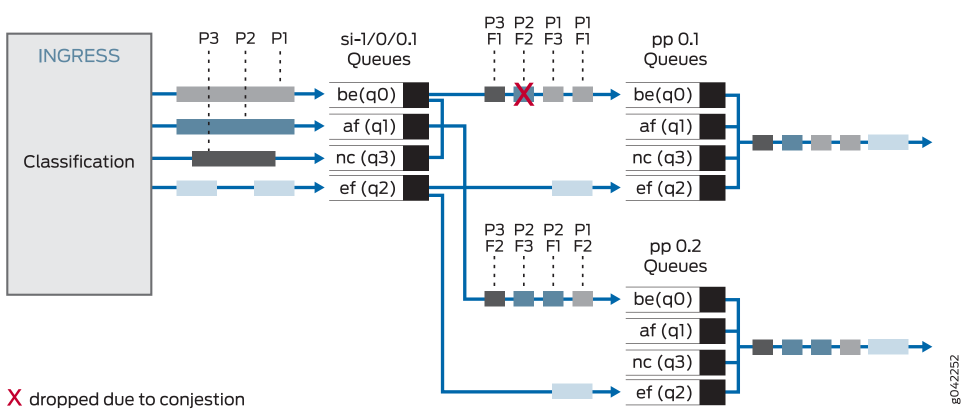
The MX Series provides two stages of queuing for packets exiting an MLPPP bundle:

- The first stage of queuing is performed at the inline services **si** interface.
- The second stage is performed by adding member link scheduler queues.

During the first stage of queuing at the **si** interface, when exiting from these queues, LFI packets are fragmented and assigned a sequence number. These fragmented packets are then distributed to the member links where they are queued for the second time.

Congestion at the member link queues can result in MLPPP packet fragments being dropped, as shown in Figure 12 on page 231. Packet flows in the figure use the notation P*x*,F*x*; for example, P1,F1 represents Packet 1, Fragment 1.

Figure 12: Dropped Sequenced Packet Fragment



Data packet and fragment P2,F2 is dropped due to congestion at the **pp0.1** queues. This occurs after the sequence numbers have been assigned for packet P2.

In a Broadband Remote Access Server (B-RAS) implementation, the bundle member links share the physical interface with other bundle member links, as well as with PPP subscriber interfaces, causing the physical interface to be oversubscribed and most likely creating congestion.

During the second stage of queuing, member link scheduler queues are added to provide a degree of protection against the port traffic congestion causing fragmented MLPPP packets to be dropped. See Figure 13 on page 232 and Figure 14 on page 232 for member link scheduler hierarchies.

NOTE: All MLPPP packets are sent to queue 0 (be).

Figure 13: si Bundle Interface Scheduler Hierarchy

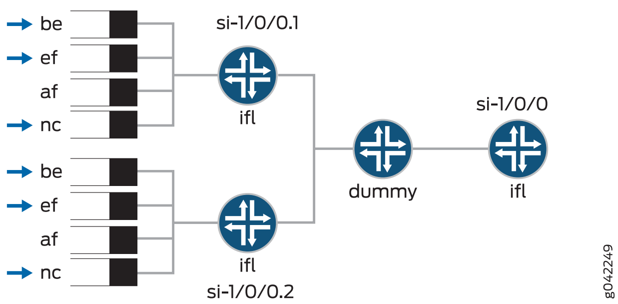
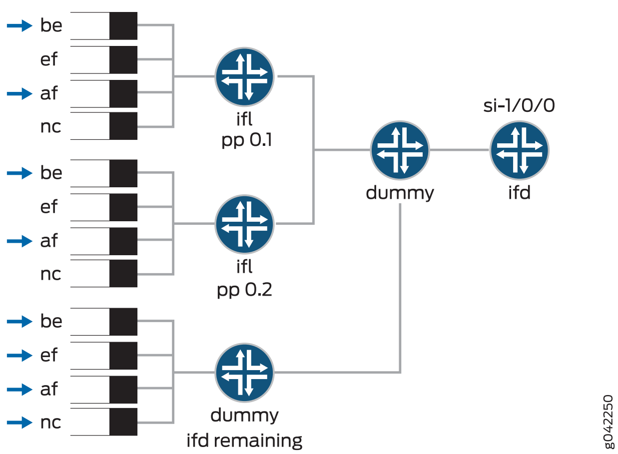


Figure 14: MLPPP Member Link Scheduler Hierarchy



Related Documentation

- Understanding MLPPP Link Fragmentation and Interleaving on page 223
- Understanding MLPPP and Fragmentation-Maps on page 224
- Understanding Fragmented Packet Queuing on page 227

CHAPTER 24

Configuring Inline Service Interfaces for LNS and PPPoE Subscribers

MLPPP Bundles and Inline Service Logical Interfaces Overview

Each MLPPP bundle for LNS or PPPoE (terminated and tunneled) subscribers is represented by an inline service (**si**) logical interface (IFL).

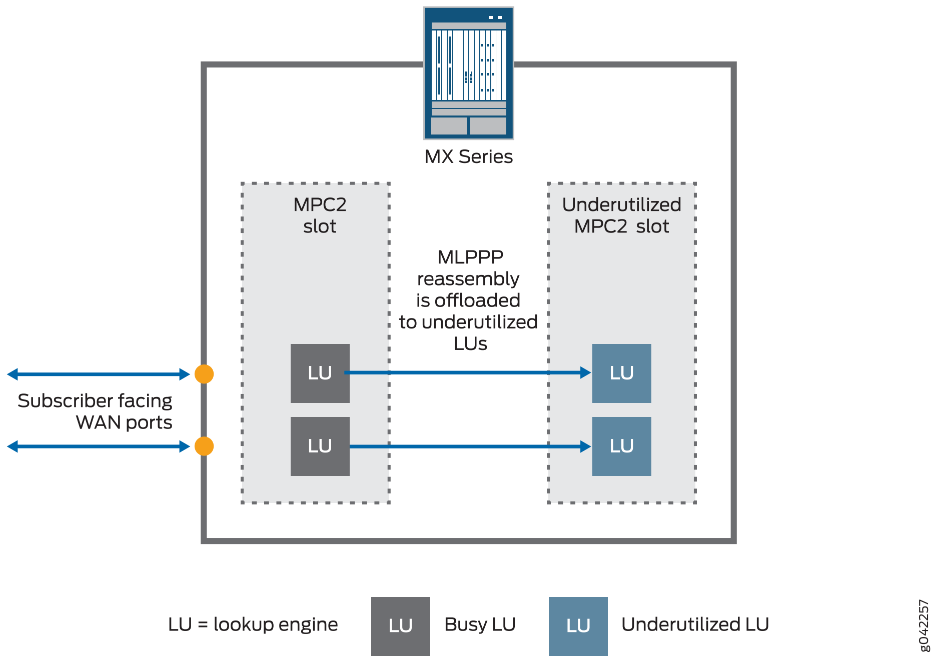
This topic contains the following sections:

Distribution of Reassembly Processing

L2TP network server (LNS) can sustain a throughput of approximately 67 percent of line rate for 64-byte packets. Additionally, MLPPP reassembly must be performed on a subset of these L2TP sessions. By introducing an **si** interface for the bundle, some of the MLPPP reassembly processing can be offloaded to another lookup engine different from the one that is performing the LNS processing.

For example, Figure 15 on page 234 shows a typical MX Series containing two access-facing MPC2 slots, with each slot containing two lookup engines. One or two of the lookup engines are underutilized within the MPC2 slots. The underutilized lookup engines are available to host **si** interfaces to offload MLPPP reassembly processing.

Figure 15: Distribution of MLPPP Reassembly Processing



NOTE: To minimize fragment reordering, the MLPPP si interface must be on an MPC2 where shaping and queuing is performed at the bundle.

Aggregation Point for True Multilink PPP

You can map each link of a multilink bundle to a different lookup engine for LNS processing. Using an **si** interface for the bundle guarantees that all fragments belonging to the same bundle arrive at a single lookup engine for reassembly.

LAC Subscriber Bundle

After a subscriber is tunneled, the bundle is no longer involved in both the control plane and the forwarding path, and both MLPPP bundle IFL and session ID are noted in the graphical user interface.

Related Documentation

- Enabling Inline Service Interfaces for PPPoE and LNS Subscribers on page 235
- Configuring Inline Service Interface for PPPoE and LNS Subscribers on page 236
- Understanding MLPPP Link Fragmentation and Interleaving on page 223

Enabling Inline Service Interfaces for PPPoE and LNS Subscribers

The inline service (**si**) interface is a virtual physical interface that resides on lookup engines. The **si** interface, referred to as an *anchor* interface, makes it possible to support multilink PPP (MLPPP) bundles without a special services PIC. The **si** interface is supported on MLPPP on the MX Series.

Four inline service interfaces are configurable per MPC-occupied chassis slot. The following MPC2 slots are supported:

- The MPC2-3D contains two lookup engines, each with two **si** interfaces.
- The PC1-3D contains only one lookup engine and it hosts all four **si** interfaces.

You can configure the following inline service interfaces as anchor interfaces for MLPPP bundles: **si-slot/0/0**, **si-slot/1/0**, **si-slot/2/0**, and **si-slot/3/0**.

- For MLPPP over PPPoE subscribers, **family mlppp** is supported in **pp0** member link IFL, and the bundle is an **si** IFL.
- For MLPPP over LNS subscribers, **family mlppp** is supported in **si-** member link IFL, and the bundle is an **si** IFL.

You enable inline services for PICs 0 to 3 individually by setting the **inline-services** statement at the **[edit chassis]** hierarchy level for the FPCs.

The following example shows how to enable inline services for PIC 0 on MPC slot 1, and PIC 1 on MPC on slot 5, and set 10g as the bandwidth for tunnel traffic. As a result, both **si-1/0/0** and **si-5/0/0** are created for the specified PICs as well.

To enable inline service interfaces:

1. Access an MPC-occupied slot and the PIC where the interface is to be enabled.

   [edit chassis]
   user@host# **edit fpc *slot-number* pic *number***

2. Enable the interface and specify the amount of bandwidth reserved on each lookup engine for tunnel traffic using inline services.

   [edit chassis fpc *slot-number* pic *number*]
   user@host# **set inline-services *bandwidth***

The following shows sample output:

```
chassis {
    fpc 1 {
        pic 0 {
            inline-services {
                bandwidth 10g;
            }
        }
    }
    fpc 5 {
        pic 1 {
```

```
            inline-services {
                bandwidth 10g;
            }
        }
    }
}
```

Related Documentation

- Configuring Inline Service Interface for PPPoE and LNS Subscribers on page 236
- Configuring Service Device Pools for Load Balancing PPPoE and LNS Subscribers on page 237
- MLPPP Bundles and Inline Service Logical Interfaces Overview on page 233

Configuring Inline Service Interface for PPPoE and LNS Subscribers

The inline service (**si**) interface is a virtual physical interface that resides on lookup engines. The **si** interface, referred to as an *anchor* interface, makes it possible to support multilink PPP (MLPPP) bundles without a special services PIC. The **si** interface is supported on MLPPP on the MX Series. Four inline service interfaces are configurable per MPC-occupied chassis slot.

For existing Layer 2 and Layer 3 services, the **si** interface **unit 0** is currently used to store the unilist next-hop information. However, you must reserve and configure **si** interface **unit 0** and set **family inet** for both PPPoE and LNS subscribers because the **si** interface implements the bundle functionality. Setting **family inet6** is ignored by the system.

The following example shows how to configure inline services for PIC 0 on MPC slot 1, and PIC 1 on MPC on slot 5, and set **unit 0 family inet** for both.

To configure inline service interfaces:

1. Access the service interface.

   [edit interfaces]
   user@host# **edit si-*slot/pic/port***

2. (Optional; for per-session shaping only) Enable the inline service interface for hierarchical schedulers and limit the number of scheduler levels to two.

   [edit interfaces si-*slot/pic/port*]
   user@host# **set hierarchical-scheduler maximum-hierarchy-levels 2**

3. (Optional; for per-session shaping only) Configure services encapsulation for inline service interface.

   [edit interfaces si-*slot/pic/port*]
   user@host# **set encapsulation generic-services**

4. Reserve and configure the IPv4 family (**inet**) on the reserved **unit 0** logical interface for PPPoE and LNS subscribers and bundle functionality.

   [edit interfaces si-*slot/pic/port*]
   user@host# **set unit 0 family inet**

The following shows sample output:

```
interfaces {
    si-1/0/0 {
        hierarchical-scheduler maximum-hierarchy-levels 2;
        encapsulation generic-services;
        unit 0 {
            family inet;
        }
    }
    si-5/1/0 {
        hierarchical-scheduler maximum-hierarchy-levels 2;
        encapsulation generic-services;
        unit 0 {
            family inet;
        }
    }
}
```

Related Documentation

- Configuring Service Device Pools for Load Balancing PPPoE and LNS Subscribers on page 237
- MLPPP Bundles and Inline Service Logical Interfaces Overview on page 233
- Enabling Inline Service Interfaces for PPPoE and LNS Subscribers on page 235

Configuring Service Device Pools for Load Balancing PPPoE and LNS Subscribers

With dynamic L2TP network server (LNS) configuration, you can replace the **services-interfaces** with a **service-device-pool** in the tunnel-group for load balancing LNS subscribers. Optionally, you can use the **service-device-pool** statement for load balancing to dynamically select the inline services (**si**) interface for both bundle (PPPoE or LNS subscribers), and LNS member link, respectively.

NOTE: **The service-device-pool configuration enables interface overlap, which can result in over usage of the overlapped interfaces.**

Before you begin, enable the inline service interfaces for all FPC slots and PICs. See "Enabling Inline Service Interfaces for PPPoE and LNS Subscribers" on page 235.

The following example shows how to configure two service device pools (pool1 and pool2) for inline services for load balancing bundle and LNS member link.

To configure two service device pools:

1. Create the tunnel group.

   [edit services l2tp]
   user@host# **set tunnel-group** ***name***

2. Define the service device pools to assign **si** interfaces for load balancing.

   [edit services l2tp]
   user@host# **set service-device-pool** ***pool-name***

The following shows sample output when all referenced FPC slots and PICs had been enabled for inline services:

```
services {
    service-device-pools {
        pool pool1 {
            interface si-1/0/0;
            interface si-1/1/0;
            interface si-3/0/0;
        }
        pool pool2 {
            interface si-1/1/0;
            interface si-2/1/0;
            interface si-5/1/0;
        }
    }
}
```

Related Documentation

- Configuring Inline Service Interface for PPPoE and LNS Subscribers on page 236
- MLPPP Bundles and Inline Service Logical Interfaces Overview on page 233
- Example: Configuring Dynamic LNS MLPPP Subscribers on page 269

CHAPTER 25

Configuring L2TP Access Client for MLPPP Subscribers

Configuring L2TP Client Access to Support MLPPP for Static Subscribers

To enable MLPPP over LT2P network server (LNS) support for MX Series, you must indicate whether MLPPP is supported for static subscribers from a particular L2TP client (LAC) by configuring the **multilink** statement currently supported in **access profile**. Access profiles define how to validate Layer 2 Tunneling Protocol (L2TP) connections and session requests. Within each L2TP access profile, you configure one or more clients (LACs). You can configure multiple access profiles and multiple clients within each profile.

With mixed mode support, the **multilink** statement enables MLPPP but does not set it. However, if you do not configure the **multilink** statement, MLPPP is not supported for static LAC subscribers.

The following two examples show L2TP access profile configurations for an MLPPP-capable static L2TP client and non-multilink (single link) static L2TP client.

To configure an L2TP access profile for MLPPP-capable static L2TP clients:

1. Create the access profile.

   [edit access]
   user@host# **edit profile *access-profile-name***

2. Configure characteristics for one or more clients (LACs).

   [edit access profile *access-profile-name*]
   user@host# **client *client-name***

3. Associate a group profile containing PPP attributes to apply for the PPP sessions being tunneled from this LAC client.

   [edit access profile *access-profile-name* client *client-name*]
   user@host# **set user-group-profile *group-profile-name***

4. Configure the LNS to renegotiate the link control protocol (LCP) with the PPP client.

   [edit access profile *access-profile-name* client *client-name*] user@host# **set l2tp lcp-renegotiation**

5. Configure the maximum number of sessions allowed in a tunnel from the client (LAC).

   [edit access profile *access-profile-name* client *client-name*]
   user@host# **set l2tp maximum-sessions-per-tunnel *number***

6. Configure the tunnel password used to authenticate the client (LAC).

   [edit access profile *access-profile-name* client *client-name*]
   user@host# **set l2tp shared-secret *shared-secret***

7. (Optional) Specify a local access profile that overrides the global access profile and the tunnel group AAA access profile to configure RADIUS server settings for the client.

   [edit access profile *access-profile-name* client *client-name*]
   user@host# **set l2tp aaa-access-profile**

8. Specify that the L2TP client is MLPPP-capable for static subscribers.

   [edit access profile *access-profile-name* client *client-name*]
   user@host# **set l2tp multilink**

MLPPP is first negotiated with static subscribers coming from the LAC peer group profile, **ce-lac-1-gp**, but then switches to PPP if the subscriber rejects MLPPP. The following shows sample output for MLPPP-capable static L2TP client:

```
access profile {
        ce-l2tp-profile1 {
        client ce-lac-1 {
            user-group-profile ce-lac-1-gp;
            l2tp {
                interface-id not-used;
                lcp-renegotiation;
                maximum-sessions-per-tunnel 2000;
                shared-secret "$9$2wgUHQF/9pB";
                aaa-access-profile ce-aaa-profile;
                multilink;
            }
        }
    }
}
```

To configure an L2TP access profile for non-MLPPP, or single link static L2TP clients, repeat Step 1 through Step 7 for configuring an L2TP access profile for multilink-capable static L2TP clients. Do not **set l2tp multilink**.

Only PPP is negotiated with static subscribers from the LAC peer group profile, **ce-lac-2-gp**, and an LCP configuration request from the customer premises equipment (CPE) with maximum received reconstructed unit (MRRU) option is rejected. The following shows sample output for single link static L2TP client:

```
access profile {
        ce-l2tp-profile1 {
        client ce-lac-2 {
            user-group-profile ce-lac-1-gp;
            l2tp {
                interface-id not-used;
                maximum-sessions-per-tunnel 1000;
                shared-secret "$9$2aBcXyz/21P";
                aaa-access-profile ce-aaa-profile;
```



```
                                                    ## multilink not entered,
static subscriber is single link only
            }
        }
    }
}
```

Related Documentation

- Mixed Mode Support for MLPPP and PPP Subscribers Overview on page 220
- MLPPP Support for LNS and PPPoE Subscribers on MX Series Overview on page 217
- Example: Configuring Static LNS MLPPP Subscribers on page 243

Configuring L2TP Client Access to Support MLPPP for Dynamic Subscribers

To enable support for MLPPP over LT2P network server (LNS) on MX Series, you configure the **family mlppp** statement in the **dynamic profile** name, which indicates that MLPPP is supported for dynamic subscribers from a particular L2TP client (LAC).

NOTE: **The multilink statement used to enable MLPPP for static LNS subscribers is ignored for dynamic LNS subscribers if it is configured.**

You can configure a dynamic profile name for the LAC using **access profile** from the **l2tp** statement. If you specify a dynamic profile name in the L2TP client access profile, it overrides the **dynamic-profile** ***name*** specified in the **tunnel-group** used to create the dynamic subscriber interface. If you do not configure a dynamic profile name in the L2TP client access profile, then the **dynamic-profile** ***name*** specified in the **tunnel-group** is used.

The following example shows an L2TP access profile configuration with a dynamic profile name for dynamic LNS subscribers.

To configure an L2TP access profile configuration with a dynamic profile name for dynamic LNS subscribers:

1. Create the access profile.

   [edit access]
   user@host# **edit profile *access-profile-name***

2. Configure characteristics for one or more clients (LACs).

   [edit access profile *access-profile-name*]
   user@host# **client *client-name***

3. Associate a group profile containing PPP attributes to apply for the PPP sessions being tunneled from this LAC client.

   [edit access profile *access-profile-name* client *client-name*]
   user@host# **set user-group-profile *group-profile-name***

4. Configure the maximum number of sessions allowed in a tunnel from the client (LAC).

   [edit access profile *access-profile-name* client *client-name*]
   user@host# **set l2tp maximum-sessions-per-tunnel *number***

5. Configure the tunnel password used to authenticate the client (LAC).

   [edit access profile *access-profile-name* client *client-name*]
   user@host# **set l2tp shared-secret** ***shared-secret***

6. (Optional) Specify a local access profile that overrides the global access profile and the tunnel group AAA access profile to configure RADIUS server settings for the client.

   [edit access profile *access-profile-name* client *client-name*]
   user@host# **set l2tp aaa-access-profile**

7. Specify the dynamic profile name for the dynamic LNS subscriber.

   [edit access profile *access-profile-name* client *client-name*]
   user@host# **set l2tp dynamic-profile** ***name***

If the **family mlppp** statement is configured in **dynamic-profile**, MLPPP is negotiated first; otherwise, only PPP is negotiated. The following shows sample output for an L2TP access profile configuration with a dynamic profile name for dynamic LNS subscribers:

```
access profile {
        ce-l2tp-profile2 {
        client ce-lac-3 {
            user-group-profile ce-lac-1-gp;
            l2tp {
                interface-id not-used;
                maximum-sessions-per-tunnel 2000;
                shared-secret "$9$2wgUHQF/9pB";
                aaa-access-profile ce-aaa-profile;
                dynamic-profile ml-lns-member-prof;
            }
        }
    }
}
```

Related Documentation

- *Configuring a Dynamic Profile for Dynamic LNS Sessions*
- Example: Configuring Dynamic LNS MLPPP Subscribers on page 269
- MLPPP Support for LNS and PPPoE Subscribers on MX Series Overview on page 217

CHAPTER 26

Configuring Static MLPPP Subscribers for MX Series

Example: Configuring Static LNS MLPPP Subscribers

This example shows how to configure static L2TP network server (LNS) multilink (MLPPP) subscribers.

Requirements

This example uses the following hardware and software components:

- MX Series with MPC2s installed
- Junos OS Release 13.3 or later

Before you configure static L2TP network server (LNS) multilink (MLPPP) subscribers, be sure you have:

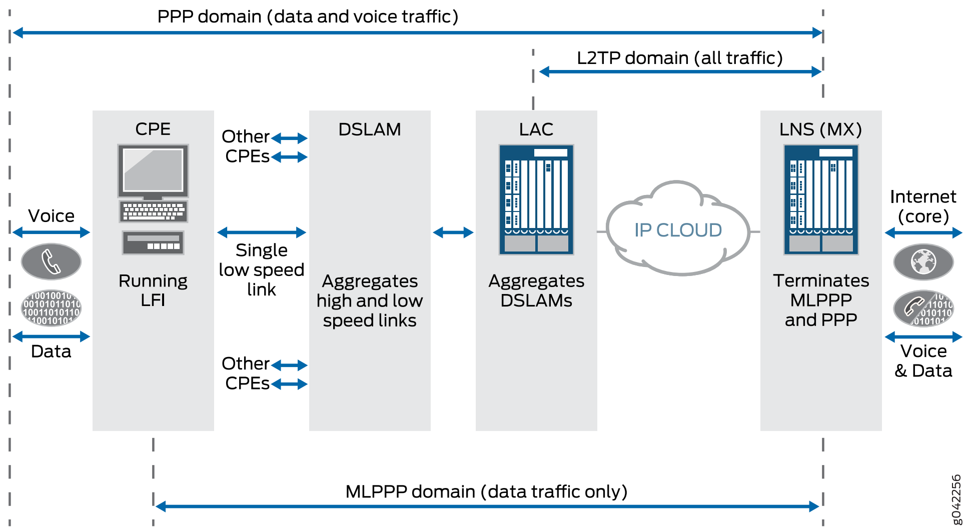
- Enabled the inline service (**si**) interface for LNS subscribers. See "Enabling Inline Service Interfaces for PPPoE and LNS Subscribers" on page 235.
- Configured the inline service (**si**) interface for LNS subscribers. See "Configuring Inline Service Interface for PPPoE and LNS Subscribers" on page 236.

Overview

An MLPPP subscriber consists of two IFLs (logical interfaces), a member link, and a bundle. For static MLPPP subscribers, you configure the member link and bundle statically. For static LNS MLPPP subscribers, you configure both member link and bundle IFLs manually. After you configure the subscriber's interface using the **family mlppp** setting, before the member link IFL can start LCP (link control protocol) negotiation for an LNS, you must also fully configure the member link's bundle IFL. Figure 16 on page 244 shows how the different types of traffic traverse through a network where the MX Series device is acting as the LNS to terminate MLPPP bundles.

Topology

Figure 16: MLPPP Bundles Terminated at MX Series as the LNS Network



The following three domains are shown passing traffic through the LNS network:

- PPP domain—Contains data and voice traffic
- MLPPP domain—Contains data traffic only
- L2TP domain—Contains all types of traffic

Configuration

To configure static L2TP network server (LNS) multilink (MLPPP) subscribers, perform these tasks:

- Configuring a Tunnel Group with Inline Service Interface and L2TP Access Profile Attributes on page 245
- Configuring a Static LNS Member Link IFL on page 247
- Configuring a Static Inline Services MLPPP Bundle IFL on page 248
- Results on page 249

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.

```
[edit]
```

```
set access profile ce-l2tp-profile1 client ce-lac-1 user-group-profile ce-lac-1-gp
set access profile ce-l2tp-profile1 client ce-lac-1 l2tp lcp-renegotiation
set access profile ce-l2tp-profile1 client ce-lac-1 l2tp maximum-sessions-per-tunnel 2000
set access profile ce-l2tp-profile1 client ce-lac-1 l2tp shared-secret "password"
set access profile ce-l2tp-profile1 client ce-lac-1 l2tp multilink
set services l2tp tunnel-group lns1 l2tp-access-profile ce-l2tp-profile1
set services l2tp tunnel-group lns1 aaa-access-profile ce-authenticator
set services l2tp tunnel-group lns1 local-gateway address 100.1.1.2
set services l2tp tunnel-group lns1 service-interface si-1/0/0

[edit]
set interfaces si-1/0/0.1
set interfaces si-1/0/0.1 dial-options l2tp-interface-id not used dedicated
set interfaces si-1/0/0.1 family mlppp bundle si-5/1/0.100
set interfaces si-1/0/0.1 family inet unnumbered-address lo0.0
set interfaces si-1/0/0.2
set interfaces si-1/0/0.2 dial-options l2tp-interface-id not used dedicated
set interfaces si-1/0/0.2 family mlppp bundle si-5/1/0.101
set interfaces si-1/0/0.2 family inet

[edit]
set interfaces si-5/0/0 unit 100
set interfaces si-5/0/0 unit 100 encapsulation multilink-ppp
set interfaces si-5/0/0 unit 100 mrru 1500
set interfaces si-5/0/0 unit 100 fragment-threshold 640
set interfaces si-5/0/0 unit 100 short-sequence
set interfaces si-5/0/0 unit 100 ppp-options dynamic-profile l2l3-service-prof
```

Configuring a Tunnel Group with Inline Service Interface and L2TP Access Profile Attributes

Step-by-Step Procedure

The following example requires that you navigate various levels in the configuration hierarchy.

To configure a tunnel group with inline service interface (**si**) and L2TP access profile attributes for static LNS MLPPP subscribers:

1. Create the access profile.

   ```
   [edit access]
   user@host# set profile ce-l2tp-profile1
   ```

2. Configure an L2TP (LAC) access client.

   ```
   [edit access profile ce-l2tp-profile1]
   user@host# set client ce-lac-1
   ```

3. Associate a group profile containing PPP attributes to apply for the PPP sessions being tunneled from this LAC client.

   ```
   [edit access profile ce-l2tp-profile1 client ce-lac1ce-lac1]
   user@host# set user-group-profile ce-lac-1-gp
   ```

4. Configure the following L2TP access profile attributes for this example:

- Link control protocol (LCP) with the PPP client.
- Maximum number of sessions allowed in a tunnel from the client (LAC).
- Tunnel password used to authenticate the client (LAC).
- L2TP client is MLPPP-capable for static subscribers. The **multilink** statement determines whether MLPPP is supported for subscribers coming in from the LAC peer.

[edit access profile ce-l2tp-profile1 client ce-lac1ce-lac1]
user@host# **set l2tp lcp-renegotiation**
user@host# **set l2tp maximum-sessions-per-tunnel 2000**
user@host# **set l2tp shared-secret *password***
user@host# **set l2tp multilink**

NOTE: Do not specify a dynamic profile name in the L2TP access client profile for static LNS MLPPP subscribers.

5. Create the tunnel group.

   [edit services l2tp]
   user@host# **set tunnel-group lns1**

6. Set the tunnel access profile equal to the setting you defined for the access profile.

   [edit services l2tp tunnel-group lns1]
   user@host# **set l2tp-access-profile ce-l2tp-profile1**

7. Set the L2TP AAA access profile.

   NOTE: You can specify the L2TP AAA access profile at either the **[edit access]** or **[edit services]** hierarchy levels, using the LNS access client profile or **tunnel-group** statements, respectively. An L2TP AAA access profile defined using the **[edit access]** hierarchy level overrides the L2TP AAA access profile defined for the **tunnel-group** using the **[edit services]** hierarchy level.

   [edit services l2tp tunnel-group lns1]
   user@host# **set aaa-access-profile ce-authenticator**

8. Set the local gateway address for the L2TP tunnel.

   [edit services l2tp tunnel-group lns1]
   user@host# **set local-gateway address 100.1.1.2**

9. Specify the inline services interface (**si**) for the static LNS MLPPP subscribers.

   [edit services l2tp tunnel-group lns1]
   user@host# **set service-interface si-1/0/0**

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

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

Configuring a Static LNS Member Link IFL

Step-by-Step Procedure

The following example requires that you navigate various levels in the configuration hierarchy.

To configure the static LNS member link IFL, you specify the static bundle using the **family mlppp** statement.

You must also configure the **family inet** statement in the subscriber (**si**) interface. The **family inet** setting enables the L2TP long route to be installed and supported for the lookup engine to steer control packets to the Routing Engine; and also enables mixed mode support, if required.

The following example shows that both PPP and MLPPP subscribers can log in successfully using the **si-1/0/0.1** interface, whereas only MLPPP subscribers can log in successfully using the **si-1/0/0.2** interface.

1. Create the **si-1/0/0.1** and **si-1/0/0.2** interfaces.

    [edit interfaces]
    user@host# **set si-1/0/0.1**
    user@host# **set si-1/0/0.2**

2. For the **si-1/0/0.1** interface, set the L2TP dial options to specify that the logical interface can host one session at a time (dedicated).

    [edit interfaces si-1/0/0.1]
    user@host# **set dial-options l2tp-interface-id not used dedicated**

3. Enable MLPPP support and configure the static bundle inline interface (IFL).

    [edit interfaces si-1/0/0.1]
    user@host# **set family mlppp bundle si-5/1/0.100**

4. Enable LNS support and mixed mode support.

    [edit interfaces si-1/0/0.1]
    user@host# **set family inet unnumbered-address lo0.0**

5. For the **si-1/0/0.2** interface, set the L2TP dial options to specify that the logical interface can host one session at a time (dedicated).

    [edit interfaces si-1/0/0.2]
    user@host# **set dial-options l2tp-interface-id not used dedicated**

6. Enable MLPPP support and configure the static bundle inline interface (IFL).

    [edit interfaces si-1/0/0.2]
    user@host# **set family mlppp bundle si-5/1/0.101**

7. Enable LNS long route support.

[edit interfaces si-1/0/0.2]
user@host# **set family inet**

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

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

Configuring a Static Inline Services MLPPP Bundle IFL

Step-by-Step Procedure

The following example requires that you navigate various levels in the configuration hierarchy.

To configure the static inline services (**si**) interface MLPPP bundle IFL, you specify the **encapsulation multilink-ppp** statement within the **si** interface. The **si** interface anchors the bundle interface.

You can also set these optional MLPPP parameters: MRRU, short sequence, and fragment-threshold. The following example shows how to configure the static (**si**) interface MLPPP bundle IFL.

1. Create the static (**si**) interface MLPPP bundle IFL **si-5/0/0** with a unit of 100.

   [edit interfaces]
   user@host# **set si-5/0/0 unit 100**

2. Configure the **encapsulation multilink-ppp** statement to enable MLPPP bundling for the **si-5/0/0.100** interface.

   [edit interfaces si-5/0/0.100]
   user@host# **set encapsulation multilink-ppp**

3. Configure the following MLPPP options for this example:

   - **mrru**—Specifies the maximum received reconstructed unit value ranging from 1500 through 4500 bytes.
   - **fragment-threshold**—Applies to all packets and forwarding classes, ranging from 128 through 16,320 bytes.
   - **short-sequence**—Determines the header format for the MLPPP. Default is **long-sequence**.

   [edit interfaces si-5/0/0.100]
   user@host# **set mrru 1500**
   user@host# **set fragment-threshold 640**
   user@host# **set short-sequence**

4. Enable support for static (**si**) interface IFL dynamic services by configuring the **ppp-options dynamic profile** setting.

   [edit interfaces si-5/0/0.100]
   user@host# **set ppp-options dynamic-profile l2l3-service-prof**

5. 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 access**, **show services**, and **show interfaces** commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
user@host# show access profile ce-l2tp-profile1
  access profile {
     ce-l2tp-profile1 {
       client ce-lac-1 {
         user-group-profile ce-lac-1-gp;
         l2tp {
           interface-id not-used;
           lcp-renegotiation;
           maximum-sessions-per-tunnel 2000;
           shared-secret "$9$2wgUHQF/9pB";
           multilink;
         }
       }
     }
  }
```

```
user@host# show services l2tp tunnel-group lns1
  services l2tp {
    tunnel-group lns1 {
      l2tp-access-profile ce-l2tp-profile1;
      aaa-access-profile ce-authenticator;
      local-gateway {
        address 100.1.1.2;
      }
      service-interface si-1/0/0;
    }
  }
```

```
user@host# show interfaces si-1/0/0
  interfaces {
    si-1/0/0 {
      unit 1 {
        dial-options {
          l2tp-interface-id not-used;
          dedicated;
        }
        family mlppp {
           bundle si-5/1/0.100;
        }
        family inet {
          unnumbered-address lo0.0;
        }
      }
      unit 2 {
        dial-options {
          l2tp-interface-id not-used;
          dedicated;
        }
```

```
            family mlppp {
              bundle si-5/1/0.101;
            }
            family inet;
          }
        }
      }
    user@host# show interfaces si-5/1/0
      interfaces {
        si-5/1/0 {
          unit 100 {
            encapsulation multilink-ppp;
            mrru 1500;
            fragment-threshold 640;
            short-sequence;
            ppp-options {
              dynamic-profile l2l3-service-prof;
            }
          }
        }
      }
```

Verification

Confirm that the configuration is working properly.

Verifying the Inline Services Interface Information

Purpose Verify that the inline services (**si**) interface is configured.

Action root@haverhill> **show interfaces si-1/0/0 extensive**

```
Physical interface: si-1/0/0, Enabled, Physical link is Up
  Interface index: 143, SNMP ifIndex: 569, Generation: 146
  Type: Adaptive-Services, Link-level type: Adaptive-Services, MTU: 9192,
  Clocking: Unspecified, Speed: 10000mbps
  Device flags   : Present Running
  Interface flags: Point-To-Point SNMP-Traps Internal: 0x4000
  Link type      : Full-Duplex
  Link flags     : None
  Physical info  : Unspecified
  Hold-times     : Up 0 ms, Down 0 ms
  Current address: Unspecified, Hardware address: Unspecified
  Alternate link address: Unspecified
  Last flapped   : Never
  Statistics last cleared: Never
  Traffic statistics:
   Input  bytes  :                 6068                    0 bps
   Output bytes  :              1072104                  352 bps
   Input  packets:                  126                    0 pps
   Output packets:                12185                    0 pps
   IPv6 transit statistics:
    Input  bytes  :                   0
    Output bytes  :                   0
    Input  packets:                   0
    Output packets:                   0
  Input errors
    Errors: 0, Drops: 0, Framing errors: 0, Runts: 0, Giants: 0,
    Policed discards: 0, Resource errors: 0
  Output errors:
    Carrier transitions: 0, Errors: 0, Drops: 0, MTU errors: 0,
    Resource errors: 0
```

Meaning The (**si**) interface is enabled with its physical link up and running with Point-to-Point interface flags set. It is shared between LNS subscribers, LNS MLPPP member links, and MX Series MLPPP bundles.

Verifying the Bundle IFL Information

Purpose Verify that the bundle IFL information is correct for MLPPP over LNS subscribers.

Action `root@haverhill>` **show interfaces si-5/1/0.1073756926 extensive**

```
Logical interface si-5/1/0.1073756926 (Index 102) (SNMP ifIndex 607)
(Generation 167)
 Flags: Up Point-To-Point SNMP-Traps 0x84000 Encapsulation: Multilink-PPP
 Last flapped: 2011-04-08 14:13:21 PDT (00:41:48 ago)
 Bandwidth: 10000mbps
 Bundle links information:
   Active bundle links       1
   Removed bundle links      0
   Disabled bundle links     0
 Bundle options:
   MRRU                                 1504
   Remote MRRU                          1504
   Drop timer period                    0
   Inner PPP Protocol field compression disabled
   Sequence number format               long (24 bits)
   Fragmentation threshold              500
   Links needed to sustain bundle       1
   Interleave fragments                 Enabled
   Multilink classes                    0
   Link layer overhead                  4.0 %
 Bundle status:
   Received sequence number        0xffffff
   Transmit sequence number        0xffffff
   Packet drops                    0 (0 bytes)
   Fragment drops                  0 (0 bytes)
   MRRU exceeded                   0
   Fragment timeout                0
   Missing sequence number         0
   Out-of-order sequence number    0
   Out-of-range sequence number    0
   Packet data buffer overflow     0
   Fragment data buffer overflow   0
 Statistics         Frames        fps         Bytes          bps
 Bundle:
   Multilink:
     Input :             3          0           270           0
     Output:             3          0           285           0
   Network:
     Input :             3          0           252           0
     Output:             3          0           276           0
 IPV6 Transit Statistics         Packets        Bytes
   Network:
     Input :                      0              0
     Output:                      0              0
 Link:
   si-1/0/0.1073756925
     Up time: 00:06:37
     Input :           126          0          9596           0
     Output:           126          0          1226           0
 Multilink detail statistics:
 Bundle:
   Fragments:
     Input :             0          0             0           0
     Output:             0          0             0           0
   Non-fragments:
     Input :             0          0             0           0
     Output:             0          0             0           0
   LFI:
```



```
       Input :             0          0             0            0
       Output:             0          0             0            0
 NCP state: inet: Opened, inet6: Not-configured, iso: Not-configured, mpls:
Not-configured
   Protocol inet, MTU: 1500, Generation: 154, Route table: 0
     Flags: Sendbcast-pkt-to-re
     Addresses, Flags: Is-Primary
       Destination: Unspecified, Local: 80.80.80.1, Broadcast: Unspecified,
Generation: 150
```

Meaning Due to the particulars of implementation, the following error counts associated with a bundle always display 0: packet drops (bytes), fragment drops (bytes), fragment timeout, missing sequence number, out-of-order sequence number, out-of-range sequence number, packet data buffer overflow and fragment data buffer overflow, and MRRU exceeded.

Verifying the Member Link IFL Information

Purpose Verify that the member link IFL information is correct for subscribers.

Action `root@haverhill>` **show interfaces si-1/0/0.1073756925 extensive**

```
  Logical interface si-5/1/0.1073756925 (Index 80) (SNMP ifIndex 3286)
    Flags: Up Point-To-Point SNMP-Traps 0x4000 Encapsulation: Adaptive-Services
    Last flapped: 2011-04-08 14:13:21 PDT (00:41:48 ago)
    Traffic statistics:
     Input  bytes  :               228
     Output bytes  :                 0
     Input  packets:                 3
     Output packets:                 0
    Local statistics:
     Input  bytes  :               228
     Output bytes  :                 0
     Input  packets:                 3
     Output packets:                 0
    Transit statistics:
     Input  bytes  :                 0                    0 bps
     Output bytes  :                 0                    0 bps
     Input  packets:                 0                    0 pps
     Output packets:                 0                    0 pps
Protocol mlppp, Multilink bundle: si-5/1/0.1073756926
Service interface: si-1/0/0, Dynamic profile: ml-bundle-prof
MTU: 9188, Generation: 15538, Route table: 0
```

Meaning Multilink bundle **si-5/1/0.1073756926** has been configured using the family **mlppp** protocol.

Verifying the Subscriber Information

Purpose Verify that the subscriber information for static MLPPP over LNS is correct.

Action root@haverhill> **show subscribers extensive**

```
  Type: L2TP
User Name: client1@rtb.com
IP Address: 80.80.80.10
IP Netmask: 255.255.255.0
Logical System: default
Routing Instance: default
Interface: si-1/0/0.1
Interface type: Static
State: Active
Radius Accounting ID: 1
Session ID: 1
Bundle Session ID: 2
Login Time: 2011-04-11 07:55:59 PDT

Type: MLPPP
User Name: client1@rtb.com
IP Address: 80.80.80.10
IP Netmask: 255.255.255.0
Logical System: default
Routing Instance: default
Interface: si-5/1/0.100
Interface type: Static
State: Active
Radius Accounting ID: 2
Session ID: 2
Underlying Session ID: 1
Login Time: 2011-04-11 07:55:59 PDT
```

Meaning Subscriber information for interface **si-5/1/0.100** has been configured for MLPPP with interface type of static.

Related Documentation

- MLPPP Support for LNS and PPPoE Subscribers on MX Series Overview on page 217
- Configuring L2TP Client Access to Support MLPPP for Static Subscribers on page 239
- Example: Configuring Static PPPoE MLPPP Subscribers on page 254

Example: Configuring Static PPPoE MLPPP Subscribers

This example shows how to configure static Point-to-Point Protocol over Ethernet (PPPoE) MLPPP for terminated and tunneled subscribers.

Requirements

This example uses the following hardware and software components:

- MX Series with MPC2s installed

- Junos OS Release 13.3 or later

Before you configure static PPPoE MLPPP for terminated and tunneled subscribers, be sure you have:

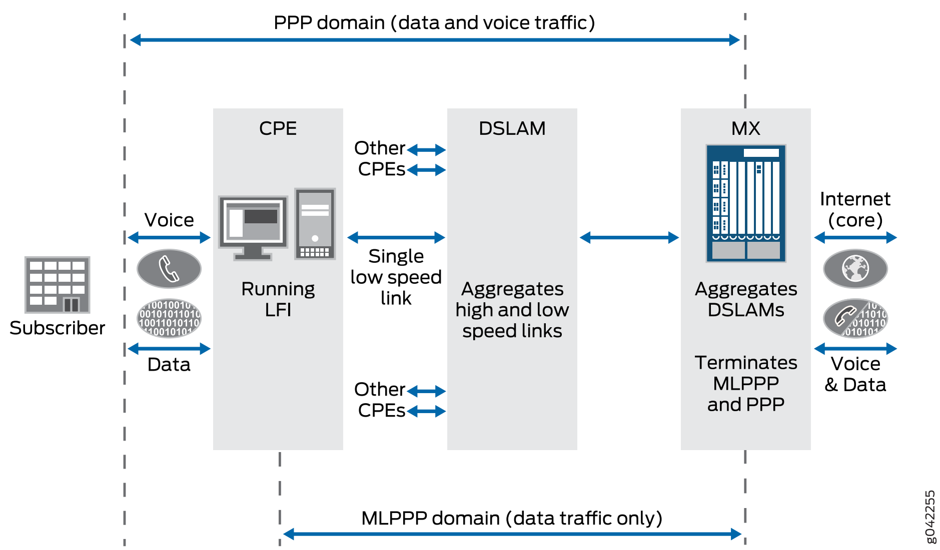
- Enabled the inline service (**si**) interface for LNS subscribers. See "Enabling Inline Service Interfaces for PPPoE and LNS Subscribers" on page 235.
- Configured the inline service (**si**) interface for LNS subscribers. See "Configuring Inline Service Interface for PPPoE and LNS Subscribers" on page 236.

Overview

An MLPPP subscriber consists of two IFLs (logical interfaces), a member link, and a bundle. For static MLPPP subscribers, you configure both member link and bundle IFLs manually. After you configure the subscriber's interface using the **family mlppp** statement, before the member link IFL can start LCP (link control protocol) negotiation PPPoE session, you must also fully configure the member link's bundle IFL. Figure 17 on page 255 shows how the different types of traffic traverse through a network where the MX Series terminates PPPoE sessions.

Topology

Figure 17: PPP and MLPPP Traffic Terminated at MX Series



The following two domains are shown terminating traffic at the MX Series:

- PPP domain—Contains data and voice traffic
- MLPPP domain—Contains data traffic only

Configuration

To configure static PPPoE MLPPP for terminated and tunneled subscribers, perform these tasks:

- Configuring a Static pp0 Member Link IFL on page 257
- Configuring a Static Inline Services MLPPP Bundle IFL on page 260
- Results on page 261

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.

```
[edit]
set interfaces ge-3/0/0 vlan-tagging
set interfaces ge-3/0/0 unit 1 encapsulation ppp-over-ether vlan-id 1
set interfaces ge-3/0/0 unit 2 encapsulation ppp-over-ether vlan-id 2
set interfaces ge-3/0/0 unit 3 encapsulation ppp-over-ether vlan-id 3
set interfaces pp0
set interfaces pp0 unit 1 keepalives interval 30
set interfaces pp0 unit 1 pppoe-options underlying interface ge-3/0/0.1 server
set interfaces pp0 unit 1 ppp-options pap chap dynamic-profile pp0-l2l3-service prof
set interfaces pp0 unit 1 family mlppp bundle si-1/0/0.1
set interfaces pp0 unit 1 family inet unnumbered-address lo0.0
set interfaces pp0 unit 1 family inet6 address 2040:2004::1.1.1.2/64
set interfaces pp0 unit 2 keepalives interval 30
set interfaces pp0 unit 2 pppoe-options underlying-interface ge-3/0/0.2 server
set interfaces pp0 unit 2 ppp-options pap dynamic-profile pp0-l2l3-service prof
set interfaces pp0 unit 2 family mlppp bundle si-1/0/0.2
set interfaces pp0 unit 3 keepalives interval 30
set interfaces pp0 unit 3 pppoe-options underlying interface ge-3/0/0.3 server
set interfaces pp0 unit 3 ppp-options pap chap dynamic-profile pp0-l2l3-service prof
set interfaces pp0 unit 3 family mlppp bundle si-1/0/0.3
set interfaces pp0 unit 3 family inet
```

```
[edit]
set interfaces si-5/0/0 unit 100
set interfaces si-5/0/0 unit 100 encapsulation multilink-ppp
set interfaces si-5/0/0 unit 100 mrru 1500
set interfaces si-5/0/0 unit 100 fragment-threshold 640
set interfaces si-5/0/0 unit 100 short-sequence
set interfaces si-5/0/0 unit 100 ppp-options dynamic-profile l2l3-service-prof
```


Configuring a Static pp0 Member Link IFL

Step-by-Step Procedure

The following example requires that you navigate various levels in the configuration hierarchy.

To configure the static PPPoE member link, you specify the static bundle using the **family mlppp** statement. PPPoE sessions are supported over the following underlying interfaces: Ethernet interfaces, static and dynamic VLAN, VLAN demultiplexing (demux) over Ethernet interfaces, and VLAN demux over aggregated Ethernet interfaces.

You must also configure the **family inet** statement in the **pp0** interface for tunneled subscribers. The **family inet** statement enables the L2TP long route to be installed and supported for the lookup engine to steer control packets to the Routing Engine.

The following example shows how to configure **pp0** member link IFL over static VLAN to support the following different types of subscribers:

- **si-1/0/0.1**—Both terminated and tunneled PPP and MLPPP subscribers can log in successfully.
- **si-1/0/0.2**—Only terminated MLPPP subscribers can log in successfully.
- **si-1/0/0.3**—Terminated and tunneled MLPPP subscribers can log in successfully.

1. Create the Gigabit Ethernet underlying interface for the PPPoE session, **ge-3/0/0**, and enable VLAN tagging.

   [edit interfaces]
   user@host# **set ge-3/0/0 vlan-tagging**

2. For the **ge-3/0/0** interface, configure PPP over Ethernet encapsulation for three VLANs.

   [edit interfaces ge-3/0/0]
   user@host# **set unit 1 encapsulation ppp-over-ether vlan-id 1**
   user@host# **set unit 2 encapsulation ppp-over-ether vlan-id 2**
   user@host# **set unit 3 encapsulation ppp-over-ether vlan-id 3**

3. Configure the dynamic PPPoE **pp0** subscriber interface to support PPPoE sessions.

   [edit interfaces]
   user@host# **set pp0**

4. Configure the first of three logical interfaces.

   a. Configure the first logical interface for the **pp0** subscriber interface on the MX Series and set an interval of 30 seconds for the keepalive value.

      [edit interfaces pp0]
      user@host# **set unit 1 keepalives interval 30**

   b. Configure the underlying interface **ge-3/0/0.1** and PPPoE server mode for a dynamic PPPoE logical interface in a dynamic profile.

      [edit interfaces pp0 unit 1]
      user@host# **set pppoe-options underlying-interface ge-3/0/0.1 server**

c. Configure PPP-specific interface properties in a dynamic profile: **pap** and **chap**, and set the **dynamic-profile** to the services dynamic profile.

NOTE: The dynamic profile is applied when Link Control Protocol (LCP) is negotiated in PPP.

[edit interfaces pp0 unit 1]
user@host# **set ppp-options pap chap dynamic-profile pp0-l2l3-service prof**

d. Configure the static bundle for the PPPoE member link for MLPPP subscribers using the **family mlppp** statement.

NOTE: The **family mlppp** statement determines whether MLPPP is supported for subscribers coming in from the underlying interface.

[edit interfaces pp0 unit 1]
user@host# **set family mlppp bundle si-1/0/0.1**

e. Configure the **family inet** statement and the unnumbered address for the protocol family required for PPP subscribers for tunneled PPP and for MLPPP subscribers.

[edit interfaces pp0 unit 1]
user@host# **set family inet unnumbered-address lo0.0**

f. (Optional) Enable the **family inet6** statement and address for the mixed mode support for PPP and MLPPP subscribers.

[edit interfaces pp0 unit 1]
user@host# **set family inet6 address 2040:2004::1.1.1.2/64**

5. Configure the second of three logical interfaces.

a. Configure the second logical interface for the **pp0** subscriber interface on the MX Series and set an interval of 30 seconds for the keepalive value.

[edit interfaces pp0]
user@host# **set unit 2 keepalives interval 30**

b. Configure the underlying interface **ge-3/0/0.2** and PPPoE server mode for a dynamic PPPoE logical interface in a dynamic profile.

[edit interfaces pp0 unit 2]
user@host# **set pppoe-options underlying interface ge-3/0/0.2 server**

c. Configure PPP-specific interface properties in a dynamic profile: **pap**, and set the **dynamic-profile** to the services dynamic profile.

NOTE: The dynamic profile is applied when Link Control Protocol (LCP) is negotiated in PPP.

[edit interfaces pp0 unit 2]
user@host# **set ppp-options pap dynamic-profile pp0-l2l3-service prof**

d. Configure the static bundle for the PPPoE member link for MLPPP subscribers using the **family mlppp** statement.

NOTE: The **family mlppp** statement determines whether MLPPP is supported for subscribers coming in from the underlying interface.

[edit interfaces pp0 unit 2]
user@host# **set family mlppp bundle si-1/0/0.2**

6. Configure the last of three logical interfaces.

a. Configure the third logical interface for the **pp0** subscriber interface on the MX Series and set an interval of 30 seconds for the keepalive value.

[edit interfaces pp0]
user@host# **set unit 3 keepalives interval 30**

b. Configure the underlying interface **ge-3/0/0.3** and PPPoE server mode for a dynamic PPPoE logical interface in a dynamic profile.

[edit interfaces pp0 unit 3]
user@host# **set pppoe-options underlying interface ge-3/0/0.3 server**

c. Configure PPP-specific interface properties in a dynamic profile: **pap** and **chap**, and set the **dynamic-profile** to the services dynamic profile.

NOTE: The dynamic profile is applied when Link Control Protocol (LCP) is negotiated in PPP.

[edit interfaces pp0 unit 3]
user@host# **set ppp-options pap chap dynamic-profile pp0-l2l3-service prof**

d. Configure the static bundle for the PPPoE member link for MLPPP subscribers using the **family mlppp** statement.

NOTE: The **family mlppp** statement determines whether MLPPP is supported for subscribers coming in from the underlying interface.

[edit interfaces pp0 unit 3]
user@host# **set family mlppp bundle si-1/0/0.3**

e. Configure tunneled subscribers.

[edit interfaces pp0 unit 3]
user@host# **set family inet**

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

[edit]
user@host# commit

Configuring a Static Inline Services MLPPP Bundle IFL

Step-by-Step Procedure

The following example requires that you navigate various levels in the configuration hierarchy.

To configure the static inline services (**si**) interface MLPPP bundle IFL, you specify the **encapsulation multilink-ppp** statement within the **si** interface. The **si** interface anchors the bundle interface.

You can also set these optional MLPPP parameters: MRRU, short sequence, and fragment-threshold. The following example shows how to configure the static **si** interface MLPPP bundle IFL:

1. Create the static (**si**) interface MLPPP bundle IFL **si-5/0/0** with a unit of 100.

   [edit interfaces]
   user@host# **set si-5/0/0 unit 100**

2. Configure the **encapsulation multilink-ppp** statement to enable MLPPP bundling for the **si-5/0/0.100** interface.

   [edit interfaces si-5/0/0.100]
   user@host# **set encapsulation multilink-ppp**

3. Configure the following MLPPP options for this example:

   - **mrru**—Specifies the maximum received reconstructed unit value ranging from 1500 through 4500 bytes.
   - **fragment-threshold**—Applies to all packets and forwarding classes, ranging from 128 through 16,320 bytes.
   - **short-sequence**—Determines the header format for the MLPPP. Default is **long-sequence**.

   [edit interfaces si-5/0/0.100]
   user@host# **set mrru 1500**
   user@host# **set fragment-threshold 640**
   user@host# **set short-sequence**

4. Enable support for static **si** interface IFL dynamic services by configuring the **ppp-options dynamic profile** statement.


```
[edit interfaces si-5/0/0.100]
user@host# set ppp-options dynamic-profile l2l3-service-prof
```

5. 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 instructions in this example to correct the configuration.

```
user@host# show interfaces ge-3/0/0
  interfaces {
      ge-3/0/0 {
      vlan-tagging;
      unit 1 {
        encapsulation ppp-over-ether;
        vlan-id 1;
      }
      unit 2 {
        encapsulation ppp-over-ether;
        vlan-id 2;
      }
      unit 3 {
        encapsulation ppp-over-ether;
        vlan-id 3;
      }
    }
    pp0 {
      unit 1 {
        keepalives interval 30;
        pppoe-options {
          underlying-interface ge-3/0/0.1;
          server;
        }
        ppp-options {
          pap;
          chap;
          dynamic-profile pp0-l2l3-service-prof;
        }
        family mlppp {
          bundle si-1/0/0.1;
        }
        family inet {
          unnumbered-address lo0.0;
        }
        family inet6 {
          address 2040:2004::1.1.1.2/64;
        }
      }
      unit 2 {
        keepalives interval 30;
```

```
            pppoe-options {
              underlying-interface ge-3/0/0.2;
              server;
            }
            ppp-options {
              pap;
              dynamic-profile pp0-l2l3-service-prof;
            }
            family mlppp {
              bundle si-1/0/0.2;
            }
          }
          unit 3 {
            keepalives interval 30;
            pppoe-options {
              underlying-interface ge-3/0/0.3;
              server;
            }
            ppp-options {
              pap;
              chap;
              dynamic-profile pp0-l2l3-service-prof;
            }
            family mlppp {
              bundle si-1/0/0.3;
            }
            family inet;
          }
        }
      }

user@host# show interfaces si-5/1/0
  interfaces {
    si-5/1/0 {
      unit 100 {
        encapsulation multilink-ppp;
        mrru 1500;
        fragment-threshold 640;
        short-sequence;
        ppp-options {
          dynamic-profile l2l3-service-prof;
        }
      }
    }
  }
```

Verification

Confirm that the configuration is working properly.

- Verifying the Bundle IFL Information on page 263
- Verifying the Member Link IFL Information on page 265
- Verifying the Subscriber Information on page 266

Verifying the Bundle IFL Information

Purpose Verify that the bundle IFL information is correct for PPPoE MLPPP subscribers.

Action `root@haverhill>` **show interfaces si-5/1/0.1073756926 extensive**

```
Logical interface si-5/1/0.1073756926 (Index 102) (SNMP ifIndex 607)
(Generation 167)
 Flags: Up Point-To-Point SNMP-Traps 0x84000 Encapsulation: Multilink-PPP
 Last flapped: 2011-04-08 14:13:21 PDT (00:41:48 ago)
 Bandwidth: 10000mbps
 Bundle links information:
   Active bundle links       1
   Removed bundle links      0
   Disabled bundle links     0
 Bundle options:
   MRRU                                 1504
   Remote MRRU                          1504
   Drop timer period                    0
   Inner PPP Protocol field compression disabled
   Sequence number format               long (24 bits)
   Fragmentation threshold              500
   Links needed to sustain bundle       1
   Interleave fragments                 Enabled
   Multilink classes                    0
   Link layer overhead                  4.0 %
 Bundle status:
   Received sequence number        0xffffff
   Transmit sequence number        0xffffff
   Packet drops                    0 (0 bytes)
   Fragment drops                  0 (0 bytes)
   MRRU exceeded                   0
   Fragment timeout                0
   Missing sequence number         0
   Out-of-order sequence number    0
   Out-of-range sequence number    0
   Packet data buffer overflow     0
   Fragment data buffer overflow   0
 Statistics          Frames        fps         Bytes          bps
 Bundle:
   Multilink:
     Input :              3          0           270           0
     Output:              3          0           285           0
   Network:
     Input :              3          0           252           0
     Output:              3          0           276           0
 IPV6 Transit Statistics         Packets        Bytes
   Network:
     Input :                      0             0
     Output:                      0             0
 Link:
   pp0.1073756925
     Up time: 00:06:37
     Input :            126          0          9596           0
     Output:            126          0          1226           0
 Multilink detail statistics:
 Bundle:
   Fragments:
     Input :              0          0             0           0
     Output:              0          0             0           0
   Non-fragments:
     Input :              0          0             0           0
     Output:              0          0             0           0
   LFI:
```



```
       Input :             0          0             0            0
       Output:             0          0             0            0
 NCP state: inet: Opened, inet6: Not-configured, iso: Not-configured, mpls:
Not-configured
   Protocol inet, MTU: 1500, Generation: 154, Route table: 0
     Flags: Sendbcast-pkt-to-re
     Addresses, Flags: Is-Primary
       Destination: Unspecified, Local: 80.80.80.1, Broadcast: Unspecified,
Generation: 150
```

Meaning Due to the particulars of implementation, the following error counts associated with a bundle always display 0: packet drops (bytes), fragment drops (bytes), fragment timeout, missing sequence number, out-of-order sequence number, out-of-range sequence number, packet data buffer overflow and fragment data buffer overflow, and MRRU exceeded.

Verifying the Member Link IFL Information

Purpose Verify that the member link IFL information is correct for subscribers.

Action `root@haverhill>` **show interfaces extensive pp0.1073756923**

```
  Logical interface pp0.1073756923 (Index 484) (SNMP ifIndex 708)
   (Generation 15544)
    Flags: Up Point-To-Point SNMP-Traps 0x4000 Encapsulation: PPPoE
    PPPoE:
      State: SessionUp, Session ID: 38,
      Session AC name: haverhill, Remote MAC address: 00:00:64:24:01:02,
      Underlying interface: ge-1/0/0.50 (Index 423)
    Bandwidth: 1000mbps
    Traffic statistics:
     Input  bytes  :                 609
     Output bytes  :                 489
     Input  packets:                  21
     Output packets:                  22
    Local statistics:
     Input  bytes  :                 133
     Output bytes  :                 377
     Input  packets:                   7
     Output packets:                   8
    Transit statistics:
     Input  bytes  :                 476                    0 bps
     Output bytes  :                 112                    0 bps
     Input  packets:                  14                    0 pps
     Output packets:                  14                    0 pps
  Keepalive settings: Interval 10 seconds, Up-count 1, Down-count 3
  LCP state: Opened
  NCP state: inet: Not-configured, inet6: Not-configured, iso: Not-configured,
mpls:
  Not-configured
  CHAP state: Success
  PAP state: Closed
    Protocol mlppp, Multilink bundle: si-1/0/0.1073756924
    Service interface: si-1/0/0, Dynamic profile: ml-bundle-service-prof
    MTU: 1526, Generation: 15535, Route table: 0
```

Meaning Logical interface **pp0.1073756923** has been configured with PPPoE, multilink bundle **si-1/0/0.1073756924**, and protocol **mlppp**.

Verifying the Subscriber Information

Purpose Verify that the subscriber information for static MLPPP over PPPoE is correct.

Action

```
root@haverhill> show subscribers detail
Type: PPPoE
User Name: user
IP Address: 10.4.1.2
IP Netmask: 255.255.0.0
Logical System: default
Routing Instance: default
Interface: pp0.20
Interface type: Static
MAC Address: 00:00:64:04:01:02
State: Active
Radius Accounting ID: 4
Session ID: 4
Bundle Session ID: 5
Login Time: 2012-02-28 10:32:24 PST

Type: MLPPP
User Name: user
IP Address: 10.4.1.2
IP Netmask: 255.255.0.0
Logical System: default
Routing Instance: default
Interface: si-1/0/0.1020
Interface type: Static
State: Active
Radius Accounting ID: 5
Session ID: 5
Underlying Session ID: 4
Login Time: 2012-02-28 10:32:24 PST
```

Meaning Subscriber information has been configured for static PPPoE with interface **pp0.20**, and static MLPPP with interface **si-1/0/0.1020**.

Related Documentation

- MLPPP Support for LNS and PPPoE Subscribers on MX Series Overview on page 217
- MLPPP Bundles and Inline Service Logical Interfaces Overview on page 233
- Example: Configuring Dynamic PPPoE MLPPP Subscribers on page 287

CHAPTER 27

Configuring Dynamic MLPPP Subscribers for MX Series

Example: Configuring Dynamic LNS MLPPP Subscribers

This example shows how to configure dynamic L2TP network server (LNS) multilink (MLPPP) subscribers.

Requirements

This example uses the following hardware and software components:

- MX Series with MPC2s installed
- Junos OS Release 13.3 or later

Before you configure dynamic LNS MLPPP subscribers, be sure you have:

- If configuring a tunnel group using an inline service (**si**) interface, enabled the inline service (**si**) interface for LNS subscribers. See "Enabling Inline Service Interfaces for PPPoE and LNS Subscribers" on page 235.
- Configured the inline service (**si**) interface for LNS subscribers. See "Configuring Inline Service Interface for PPPoE and LNS Subscribers" on page 236.
- If configuring a tunnel group using a pool of service interfaces, configured service device pools for LNS subscribers. See "Configuring Service Device Pools for Load Balancing PPPoE and LNS Subscribers" on page 237.

Overview

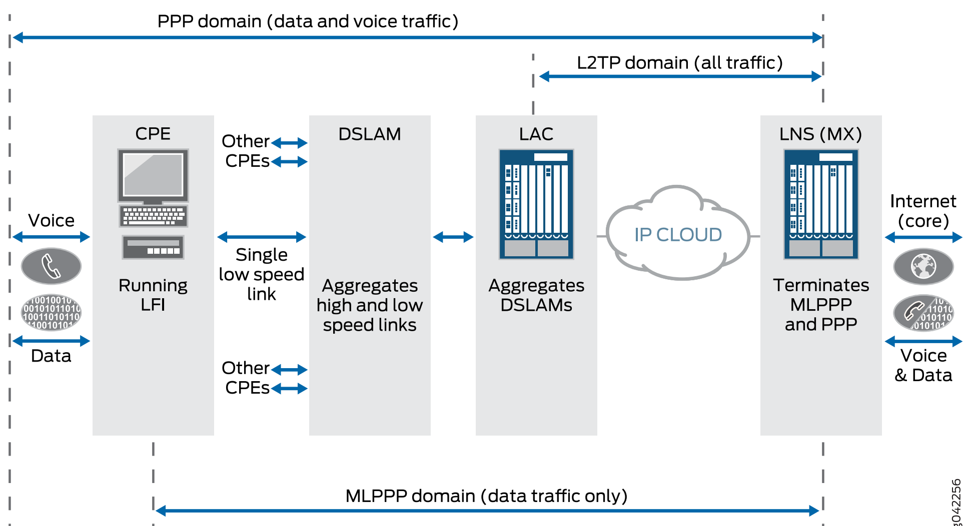
An MLPPP subscriber consists of two IFLs (logical interfaces), a member link, and a bundle. For dynamic LNS MLPPP subscribers, you configure the dynamic member link IFLs using dynamic profiles. The member link dynamic profile includes the **family mlppp** statement containing the bundle dynamic profile and the service interface (**si**), or a pool of service interfaces. This information is then used to create the dynamic bundle IFL.

Each dynamic bundle accepts only one dynamic member link. If more than one dynamic member link attempts to join the same dynamic bundle, the system fails the new member session.

Figure 18 on page 270 shows how the different types of traffic traverse through a network where the MX Series is acting as the LNS to terminate MLPPP bundles.

Topology

Figure 18: MLPPP Bundles Terminated at MX Series as the LNS Network



The following three domains are shown passing traffic through the LNS network:

- PPP domain—Contains data and voice traffic
- MLPPP domain—Contains data traffic only
- L2TP domain—Contains all types of traffic

Configuration

To configure dynamic LNS MLPPP subscribers, perform these tasks:

- Configuring a Tunnel Group with a Pool of Service Interfaces and L2TP Access Profile Attributes on page 273
- Configuring a Dynamic Profile for Dynamic LNS Member Link IFL Without Mixed Mode Support on page 275
- Configuring a Dynamic Profile for Dynamic LNS Member Link IFL With Mixed Mode Support on page 277
- Configuring a Dynamic Profile for the Dynamic Bundle IFL on page 279
- Results on page 281

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.

```
[edit]
set access profile ce-l2tp-profile2 client ce-lac-3 user-group-profile ce-lac-1-gp
set access profile ce-l2tp-profile2 client ce-lac-3 l2tp multilink
set access profile ce-l2tp-profile2 client ce-lac-3 l2tp maximum-sessions-per-tunnel
  2000
set access profile ce-l2tp-profile2 client ce-lac-3 l2tp shared-secret “password”
set access profile ce-l2tp-profile2 client ce-lac-3 l2tp dynamic-profile ml-lns-member-prof
set services l2tp tunnel-group dyn-l2tp-tunnel-group l2tp-access-profile ce-l2tp-profile2
set services l2tp tunnel-group dyn-l2tp-tunnel-group aaa-access-profile ce-authenticator
set services l2tp tunnel-group dyn-l2tp-tunnel-group local-gateway address 31.1.1.1
set services l2tp tunnel-group dyn-l2tp-tunnel-group service-device-pool pool1
set services l2tp tunnel-group dyn-l2tp-tunnel-group dynamic-profile ml-lns-member-prof

[edit]
set dynamic-profiles mlp-lns-member-profile
set dynamic-profiles mlp-lns-member-profile interfaces “$junos-interface-ifd-name”
  unit “$junos-interface-unit”
set dynamic-profiles mlp-lns-member-profile interfaces “$junos-interface-ifd-name”
  unit “$junos-interface-unit” dial-options l2tp-interface-id dont care dedicated
set dynamic-profiles mlp-lns-member-profile interfaces “$junos-interface-ifd-name”
  unit “$junos-interface-unit” family mlppp bundle “$junos-bundle-interface-name”
set dynamic-profiles mlp-lns-member-profile interfaces “$junos-interface-ifd-name”
  unit “$junos-interface-unit” family mlppp service-device-pool pool1
set dynamic-profiles mlp-lns-member-profile interfaces “$junos-interface-ifd-name”
  unit “$junos-interface-unit” family mlppp dynamic-profile ml-bundle-prof
set dynamic-profiles mlp-lns-member-profile interfaces “$junos-interface-ifd-name”
  unit “$junos-interface-unit” family inet

[edit]
set dynamic-profiles ml-bundle-prof
set dynamic-profiles ml-bundle-prof routing-instances "$junos-routing-instance" interface
  "$junos-interface-name"
```

```
set dynamic-profiles ml-bundle-prof routing-instances "$junos-routing-instance" interface
  "$junos-interface-name" routing-options access route
  $junos-framed-route-ip-address-prefix
set dynamic-profiles ml-bundle-prof routing-instances "$junos-routing-instance" interface
  "$junos-interface-name" routing-options access route
  $junos-framed-route-ip-address-prefix next-hop $junos-framed-route-nexthop
set dynamic-profiles ml-bundle-prof routing-instances "$junos-routing-instance" interface
  "$junos-interface-name" routing-options access route
  $junos-framed-route-ip-address-prefix metric $junos-framed-route-cost
set dynamic-profiles ml-bundle-prof routing-instances "$junos-routing-instance" interface
  "$junos-interface-name" routing-options access route
  $junos-framed-route-ip-address-prefix preference $junos-framed-route-distance
set dynamic-profiles ml-bundle-prof routing-instances "$junos-routing-instance" interface
  "$junos-interface-name" access-internal route $junos-subscriber-ip-address
set dynamic-profiles ml-bundle-prof routing-instances "$junos-routing-instance" interface
  "$junos-interface-name" access-internal route $junos-subscriber-ip-address
  qualified-next-hop $junos-interface-name
set dynamic-profiles ml-bundle-prof interfaces “$junos-interface-ifd-name” unit
  “$junos-interface-unit”
set dynamic-profiles ml-bundle-prof interfaces “$junos-interface-ifd-name” unit
  “$junos-interface-unit” encapsulation multilink-ppp
set dynamic-profiles ml-bundle-prof interfaces “$junos-interface-ifd-name” unit
  “$junos-interface-unit” mrru 1500
set dynamic-profiles ml-bundle-prof interfaces “$junos-interface-ifd-name” unit
  “$junos-interface-unit” short-sequence
set dynamic-profiles ml-bundle-prof interfaces “$junos-interface-ifd-name” unit
  “$junos-interface-unit” fragment-threshold 320
set dynamic-profiles ml-bundle-prof interfaces “$junos-interface-ifd-name” unit
  “$junos-interface-unit” family inet
set class-of-service traffic-control-profiles tcp2
set dynamic-profiles ml-bundle-prof class-of-service traffic-control-profiles tcp2
  scheduler-map "$junos-cos-scheduler-map”
set dynamic-profiles ml-bundle-prof class-of-service traffic-control-profiles tcp2
  shaping-rate "$junos-cos-shaping-rate"
set dynamic-profiles ml-bundle-prof class-of-service traffic-control-profiles tcp2
  guaranteed-rate "$junos-cos-guaranteed-rate"
set dynamic-profiles ml-bundle-prof class-of-service traffic-control-profiles tcp2
  delay-buffer-rate “$junos-cos-delay-buffer-rate”
set dynamic-profiles ml-bundle-prof class-of-service interfaces
  “$junos-interface-ifd-name” unit "$junos-interface-unit"
set dynamic-profiles ml-bundle-prof class-of-service interfaces
  “$junos-interface-ifd-name” unit "$junos-interface-unit" output-traffic-control-profile
  tcp2
set dynamic-profiles ml-bundle-prof class-of-service interfaces
  “$junos-interface-ifd-name” unit "$junos-interface-unit" fragmentation-map fragmap-2
```


Configuring a Tunnel Group with a Pool of Service Interfaces and L2TP Access Profile Attributes

Step-by-Step Procedure

The following example requires that you navigate various levels in the configuration hierarchy.

To configure a tunnel group with a pool of service interfaces and L2TP access profile attributes for dynamic LNS MLPPP subscribers:

1. Create the access profile.

   [edit access]
   user@host# **set profile ce-l2tp-profile2**

2. Configure an L2TP (LAC) access client.

   [edit access profile ce-l2tp-profile2]
   user@host# **set client ce-lac-3**

3. Associate a group profile containing PPP attributes to apply for the PPP sessions being tunneled from this LAC client.

   [edit access profile ce-l2tp-profile2 client ce-lac-3]
   user@host# **set user-group-profile ce-lac-1-gp**

4. Configure the following L2TP access profile attributes for this example:

   - L2TP client is multilink (MLPPP)-capable for subscribers. The **multilink** statement in the L2TP access client profile determines whether MLPPP is supported for subscribers coming in from the LAC peer.
   - Maximum number of sessions allowed in a tunnel from the client (LAC).
   - Tunnel password used to authenticate the client (LAC).
   - Dynamic profile name in the L2TP access client profile for dynamic LNS MLPPP subscribers.

   NOTE: If the **dynamic-profile** ***name*** is defined in the L2TP access client profile, it is used to create the dynamic LNS MLPPP member link; otherwise, the **dynamic-profile** ***name*** defined in the tunnel group is used. If neither profile contains the **family mlppp** statement, then the incoming LNS session fails.

   [edit access profile ce-l2tp-profile2 client ce-lac-3]
   user@host# **set l2tp multilink**
   user@host# **set l2tp maximum-sessions-per-tunnel 2000**
   user@host# **set l2tp shared-secret *password***
   user@host# **set dynamic-profile ml-lns-member-prof**

5. Create the tunnel group.

   [edit services l2tp]

user@host# **set tunnel-group dyn-l2tp-tunnel-group**

6. Set the tunnel access profile equal to the setting you defined for the access profile.

   [edit services l2tp tunnel-group dyn-l2tp-tunnel-group]
   user@host# **set l2tp-access-profile ce-l2tp-profile2**

7. Set the L2TP AAA access profile.

   NOTE: You can specify the L2TP AAA access profile in either the **[edit access]** or **[edit services]** hierarchy levels, using the LNS access client profile or **tunnel-group** statements, respectively. An L2TP AAA access profile defined using the **[edit access]** hierarchy level overrides the L2TP AAA access profile defined for the **tunnel-group** using the **[edit services]** hierarchy level.

   [edit services l2tp tunnel-group dyn-l2tp-tunnel-group]
   user@host# **set aaa-access-profile ce-authenticator**

8. Set the local gateway address for the L2TP tunnel.

   [edit services l2tp tunnel-group dyn-l2tp-tunnel-group]
   user@host# **set local-gateway address 31.1.1.1**

9. Specify the pool of service interfaces for the dynamic LNS MLPPP subscribers.

   [edit services l2tp tunnel-group dyn-l2tp-tunnel-group]
   user@host# **set service-device-pool pool1**

10. Specify the dynamic profile used to create the dynamic LNS MLPPP member link.

    [edit services l2tp tunnel-group dyn-l2tp-tunnel-group]
    user@host# **set dynamic-profile ml-lns-member-prof**

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

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

Configuring a Dynamic Profile for Dynamic LNS Member Link IFL Without Mixed Mode Support

Step-by-Step Procedure

The following example requires that you navigate various levels in the configuration hierarchy.

You can configure the **dynamic-profile *name*** used to create the dynamic LNS member link IFL in either the L2TP client access profile or in the tunnel-group. See "Configuring a Tunnel Group with a Pool of Service Interfaces and L2TP Access Profile Attributes" on page 273.

The following example shows **dynamic-profile** configuration for LNS MLPPP and PPP subscribers. The **family mlppp** statement contains the **dynamic-profile *name***, and either the **service-interface** or the **service-device-pool**, used to create the dynamic bundle IFL. If you configure a **service-device-pool**, an inline services (**si**) interface is selected from the pool to create the dynamic bundle IFL using a round-robin method.

You must also configure the **family inet** statement in the **si** member link dynamic profile interface for tunneled subscribers. The **family inet** statement enables the L2TP long route to be installed and supported for the lookup engine to steer control packets to the Routing Engine.

NOTE: **Optionally, you can configure the dynamic profile to support mixed mode to enable PPP subscribers to successfully log in using the dynamic profile. See "Configuring a Dynamic Profile for Dynamic LNS Member Link IFL With Mixed Mode Support" on page 277 for the additional configuration commands required.**

1. Specify the dynamic profile that you used to create the dynamic LNS MLPPP member link previously in "Configuring a Tunnel Group with a Pool of Service Interfaces and L2TP Access Profile Attributes" on page 273.

   ```
   [edit dynamic-profiles]
   user@host# set ml-lns-member-prof
   ```

2. Configure the interface for the dynamic profile by setting the predefined dynamic interface variable *$junos-interface-ifd-name*, and the logical interface unit by setting the predefined unit number variable *$junos-interface-unit*. The interface and unit number variables are dynamically replaced with the interface and unit number that the subscriber accesses when connecting to the MX Series.

NOTE: The interface setting for a dynamic profile for PPPoE sessions can use either of the following code formats:

- **set interfaces pp0**

  or

- **set interfaces "$junos-interface-ifd-name"**

This example uses **set interfaces "$junos-interface-ifd-name"**.

[edit dynamic-profiles ml-lns-member-prof]
user@host# **set interfaces "$junos-interface-ifd-name" unit "$junos-interface-unit"**

3. For the *$junos-interface-ifd-name* interface, set the L2TP interface dial options to specify that the logical interface can host one session at a time (dedicated).

   [edit dynamic-profiles ml-lns-member-prof interfaces "$junos-interface-ifd-name" unit "$junos-interface-unit"]
   user@host# **set dial-options l2tp-interface-id dont care dedicated**

4. Enable MLPPP support for LNS MLPPP subscribers and configure the dynamic bundle interface (IFL) by setting the predefined dynamic bundle interface variable *$junos-bundle-interface-name*.

   NOTE: The **family mlppp** statement determines whether MLPPP is supported for subscribers coming in from the underlying interface.

   [edit dynamic-profiles ml-lns-member-prof interfaces "$junos-interface-ifd-name" unit "$junos-interface-unit"]
   user@host# **set family mlppp bundle "$junos-bundle-interface-name"**

5. Specify the pool of service interfaces for the dynamic LNS MLPPP subscribers.

   [edit dynamic-profiles ml-lns-member-prof interfaces "$junos-interface-ifd-name" unit "$junos-interface-unit" family mlppp]
   user@host# **set service-device-pool pool1**

6. Specify the dynamic profile name for the bundle.

   [edit dynamic-profiles ml-lns-member-prof interfaces "$junos-interface-ifd-name" unit "$junos-interface-unit" family mlppp]
   user@host# **set dynamic-profile ml-bundle-prof**

7. Enable support for LNS subscribers and the LNS long route.

   [edit dynamic-profiles ml-lns-member-prof interfaces "$junos-interface-ifd-name" unit "$junos-interface-unit"]
   user@host# **set family inet**

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

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

Configuring a Dynamic Profile for Dynamic LNS Member Link IFL With Mixed Mode Support

Step-by-Step Procedure

The following example requires that you navigate various levels in the configuration hierarchy.

Optionally, you can configure the dynamic profile to support mixed mode to enable PPP subscribers to successfully log in using the dynamic profile.

The following example shows the additional configurations required to support mixed mode for dynamic profiles.

NOTE: The following configuration commands are not included in the "CLI Quick Configuration" on page 271 section.

1. Specify the dynamic profile that you used to create the dynamic LNS MLPPP member link previously in "Configuring a Tunnel Group with a Pool of Service Interfaces and L2TP Access Profile Attributes" on page 273.

   [edit dynamic-profiles]
   user@host# **set ml-lns-member-prof**

2. When the customer premises equipment (CPE) is for a dynamic virtual routing and forwarding (VRF) PPP subscriber, you must configure the routing instance and its interface.

   [edit dynamic-profiles ml-lns-member-prof]
   user@host# **set routing-instances "$junos-routing-instance" interface "$junos-interface-name"**

3. Configure the access route for the routing options.

   [edit dynamic-profiles ml-lns-member-prof routing-instances "$junos-routing-instance" interface "$junos-interface-name"]
   user@host# **set routing-options access route $junos-framed-route-ip-address-prefix**

4. Configure the next-hop, metric, and preference for the router.

   [edit dynamic-profiles ml-lns-member-prof routing-instances "$junos-routing-instance" interface "$junos-interface-name" routing-options access route $junos-framed-route-ip-address-prefix]
   user@host# **set next-hop $junos-framed-route-nexthop**
   user@host# **set metric $junos-framed-route-cost**
   user@host# **set preference $junos-framed-route-distance**

5. Configure the internal access route for the routing options.

   [edit dynamic-profiles ml-lns-member-prof routing-instances "$junos-routing-instance" interface "$junos-interface-name"]

user@host# **set routing-options access-internal route $junos-subscriber-ip-address**

6. Configure the qualified next-hop for the internal route..

   [edit dynamic-profiles ml-lns-member-prof routing-instances "$junos-routing-instance" interface "$junos-interface-name" routing-options access-internal route $junos-subscriber-ip-address]
   user@host# **set qualified-next-hop $junos-interface-name**

7. Follow the procedure described in "Configuring a Dynamic Profile for Dynamic LNS Member Link IFL Without Mixed Mode Support" on page 275 to configure the basic settings for the dynamic profile.

   NOTE: To enable mixed mode support, when the CPE is a PPP subscriber, you must also add an unnumbered address, and input and output filters to the **family inet** statement.

   [edit dynamic-profiles ml-lns-member-prof interfaces "$junos-interface-ifd-name" unit "$junos-interface-unit"]
   user@host# **set family inet unnumbered-address $junos-loopback-interface**
   user@host# **set family inet filter input "$junos-input-filter" output "$junos-output-filter"**

8. When the CPE is a PPP subscriber, you must also configure class of service and define the traffic control profile.

   [edit dynamic-profiles ml-lns-member-prof class-of-service]
   user@host# **set traffic-control-profiles tc-profile**

9. For the traffic-control profile, define the following settings: scheduler map, shaping rate, overhead accounting, guaranteed rate, and delay buffer rate.

   [edit dynamic-profiles ml-lns-member-prof class-of-service traffic-control-profiles tc-profile]
   user@host# **set scheduler-map "$junos-cos-scheduler-map"**
   user@host# **set shaping-rate "$junos-cos-shaping-rate"**
   user@host# **set overhead-accounting "$junos-cos-shaping-mode" bytes "$junos-cos-byte-adjust"**
   user@host# **set guaranteed-rate "$junos-cos-guaranteed-rate"**
   user@host# **set delay-buffer-rate "$junos-cos-delay-buffer-rate"**

10. Configure the interface for the dynamic profile by setting the predefined dynamic interface variable *$junos-interface-ifd-name*, and the logical interface unit by setting the predefined unit number variable *$junos-interface-unit*.

    [edit dynamic-profiles ml-lns-member-prof class-of-service]
    user@host# **set interfaces "$junos-interface-ifd-name" unit "$junos-interface-unit"**

11. For the dynamic profile interface, define the following settings: output traffic control profile, classifiers, and rewrite rules.

[edit dynamic-profiles ml-lns-member-prof class-of-service interfaces "$junos-interface-ifd-name" unit "$junos-interface-unit"]
user@host# **set output-traffic-control-profile tc-profile**
user@host# **set classifiers dscp GEN-CLASSIFIER-IN**
user@host# **set rewrite-rules dscp GEN-RW-OUT-DSCP**

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

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

Configuring a Dynamic Profile for the Dynamic Bundle IFL

Step-by-Step Procedure

The following example requires that you navigate various levels in the configuration hierarchy.

To configure the dynamic profile for the dynamic bundle IFL, you specify the **encapsulation multilink-ppp** statement within the dynamic profile. The **dynamic profile** for the dynamic bundle IFL is referenced from the **dynamic profile** for dynamic PPPoE and LNS member link IFLs.

You must configure the **fragmentation-maps** statement statically using class-of-service and assign them in the bundle dynamic profile. You can also set these optional MLPPP parameters: MRRU, short sequence, and fragment-threshold. The following example shows how to configure the dynamic profile for the dynamic bundle IFL.

1. Specify the dynamic profile name for the bundle.

   [edit dynamic-profiles}
   user@host# **set ml-bundle-prof**

2. Although MLPPP member links process authentication and routing-instance assignments, if a non-default routing-instance is assigned, you must configure the bundle IFL under the assigned routing-instance. As a result, you must also configure routing-instances in the bundle dynamic-profile.

   [edit dynamic-profiles ml-bundle-prof]
   user@host# **set routing-instances "$junos-routing-instance" interface "$junos-interface-name"**

3. Configure the access route for the routing options.

   [edit dynamic-profiles ml-bundle-prof routing-instances "$junos-routing-instance" interface "$junos-interface-name"]
   user@host# **set routing-options access route $junos-framed-route-ip-address-prefix**

4. Configure the next-hop, metric, and preference for the router.

   [edit dynamic-profiles ml-bundle-prof routing-instances "$junos-routing-instance" interface "$junos-interface-name" routing-options access route $junos-framed-route-ip-address-prefix]
   user@host# **set next-hop $junos-framed-route-nexthop**
   user@host# **set metric $junos-framed-route-cost**
   user@host# **set preference $junos-framed-route-distance**

5. Configure the internal access route for the routing options.

   [edit dynamic-profiles ml-bundle-prof routing-instances "$junos-routing-instance" interface "$junos-interface-name"]
   user@host# **set routing-options access-internal route $junos-subscriber-ip-address**

6. Configure the qualified next-hop for the internal route.

   [edit dynamic-profiles ml-bundle-prof routing-instances "$junos-routing-instance" interface "$junos-interface-name" routing-options access-internal route $junos-subscriber-ip-address]
   user@host# **set qualified-next-hop $junos-interface-name**

7. Configure the interface for the dynamic profile by setting the predefined dynamic interface variable *$junos-interface-ifd-name*, and the logical interface unit by setting the predefined unit number variable *$junos-interface-unit*. The interface and unit number variables are dynamically replaced with the interface and unit number that the subscriber accesses when connecting to the MX Series.

   [edit dynamic-profiles ml-bundle-prof]
   user@host# **set interfaces "$junos-interface-ifd-name" unit "$junos-interface-unit"**

8. Configure the **encapsulation multilink-ppp** statement to enable MLPPP bundling for the dynamic profile.

   [edit dynamic-profiles ml-bundle-prof interfaces "$junos-interface-ifd-name" unit "$junos-interface-unit"]
   user@host# **set encapsulation multilink-ppp**

9. Configure the following MLPPP options for this example:

   - **mrru**—Specifies the maximum received reconstructed unit value ranging from 1500 through 4500 bytes.
   - **fragment-threshold**—Applies to all packets and forwarding classes, ranging from 128 through 16,320 bytes.
   - **short-sequence**—Determines the header format for the MLPPP. Default is **long-sequence**.

   [edit dynamic-profiles ml-bundle-prof interfaces "$junos-interface-ifd-name" unit "$junos-interface-unit"]
   user@host# **set mrru 1500**
   user@host# **set fragment-threshold 320**
   user@host# **set short-sequence**

10. Enable support for MLPP subscribers.

    [edit dynamic-profiles ml-bundle-prof interfaces "$junos-interface-ifd-name" unit "$junos-interface-unit"]
    user@host# **set family inet**

11. To enable **fragmentation-maps** support, you must configure class-of-service and define the traffic control profile.

    [edit dynamic-profiles ml-bundle-prof class-of-service]
    user@host# **set traffic-control-profiles tcp2**

12. For the traffic-control profile, define the following settings: scheduler map, shaping rate, guaranteed rate, and delay buffer rate.

    [edit dynamic-profiles ml-bundle-prof class-of-service traffic-control-profiles tcp2]
    user@host# **set scheduler-map "$junos-cos-scheduler-map"**
    user@host# **set shaping-rate "$junos-cos-shaping-rate"**
    user@host# **set guaranteed-rate "$junos-cos-guaranteed-rate"**
    user@host# **set delay-buffer-rate "$junos-cos-delay-buffer-rate"**

13. Configure the underlying interface for the dynamic profile by setting the predefined dynamic interface variable *$junos-interface-ifd-name*, and the logical interface unit by setting the predefined unit number variable *$junos-interface-unit*. The interface and unit number variables are dynamically replaced with the interface and unit number that the subscriber accesses when connecting to the MX Series.

    [edit dynamic-profiles ml-bundle-prof class-of-service]
    user@host# **set interfaces "$junos-interface-ifd-name" unit "$junos-interface-unit"**

14. For the dynamic profile interface, define the output traffic control profile.

    [edit dynamic-profiles ml-bundle-prof class-of-service interfaces "$junos-interface-ifd-name" unit "$junos-interface-unit"]
    user@host# **set output-traffic-control-profile tcp2**

15. Define the fragmentation-map required for dynamic profile bundles and used to enable link fragmentation and interleaving (LFI).

    [edit dynamic-profiles ml-bundle-prof class-of-service interfaces "$junos-interface-ifd-name" unit "$junos-interface-unit"]
    user@host# **set fragmentation-map fragmap-2**

16. 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 access**, **show services**, and **show dynamic-profiles** commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
user@host# show access profile ce-l2tp-profile2
  access profile {
    ce-l2tp-profile2 {
      client ce-lac-3 {
        user-group-profile ce-lac-1-gp;
        l2tp {
          multilink;
          interface-id not-used;
          maximum-sessions-per-tunnel 2000;
          shared-secret "$9$2wgUHQF/9pB";
          dynamic-profile ml-lns-member-prof;
        }
      }
```

```
      }
    }
  user@host# show services l2tp tunnel-group dyn-l2tp-tunnel-group
    services {
      l2tp {
        tunnel-group dyn-l2tp-tunnel-group {
          l2tp-access-profile ce-l2tp-profile2;
          aaa-access-profile ce-authenticator;
          local-gateway {
            address 31.1.1.1;
          }
          service-device-pool pool1;
          dynamic-profile ml-lns-member-prof;
        }
      }
    }
```

Dynamic profile for dynamic LNS member link IFL without mixed mode:

```
  user@host# show dynamic-profiles mlp-lns-member-profile
    dynamic-profile mlp-lns-member-profile {
      interface $junos-interface-ifd-name {
        unit $junos-interface-unit" {
          dial-options {
            l2tp-interface-id dont-care;
            dedicated;
          }
          family mlppp {
            bundle $junos-bundle-interface-name ;
            service-device-pool pool1;
            dynamic-profile mlp-bundle-profile;
          }
          family inet {
          }
        }
      }
    }
```

Dynamic profile for dynamic LNS member link IFL with mixed mode:

```
  user@host# show dynamic-profiles mlp-lns-member-profile
    dynamic-profile ml-lns-member-prof {
     routing-instances {
        "$junos-routing-instance" {
          interface "$junos-interface-name";
            routing-options {
              access {
                route $junos-framed-route-ip-address-prefix {
                  next-hop $junos-framed-route-nexthop;
                  metric $junos-framed-route-cost;
                  preference $junos-framed-route-distance;
                }
              }
              access-internal {
                route $junos-subscriber-ip-address {
                  qualified-next-hop $junos-interface-name;
```



```
              }
            }
          }
        }
      }
    }
  interfaces "$junos-interface-ifd-name" {
      unit "$junos-interface-unit" {
        dial-options {
          l2tp-interface-id l2tp-encapsulation;
          dedicated;
        }
        family mlppp {
          bundle $junos-bundle-interface-name;
          service-device-pool pool2;
          dynamic-profile ml-bundle-prof;
        }
        family inet {
          unnumbered-address $junos-loopback-interface;
          filter {
            input "$junos-input-filter";
            output "$junos-output-filter";
          }
        }
      }
   class-of-service {
      traffic-control-profiles {
        tc-profile {
          scheduler-map "$junos-cos-scheduler-map";
          shaping-rate "$junos-cos-shaping-rate";
        overhead-accounting "$junos-cos-shaping-mode" bytes "$junos-cos-byte-adjust";

          guaranteed-rate "$junos-cos-guaranteed-rate";
          delay-buffer-rate "$junos-cos-delay-buffer-rate";
        }
      }
      interfaces {
        "$junos-interface-ifd-name" {
          unit "$junos-interface-unit" {
            output-traffic-control-profile tc-profile;
            classifiers {
              dscp GEN-CLASSIFIER-IN;
            }
            rewrite-rules {
              dscp GEN-RW-OUT-DSCP;
            }
          }
        }
      }
    }
  }
```

```
user@host# show dynamic-profiles ml-bundle-prof
  dynamic-profile ml-bundle-prof {
      routing-instances {
      "$junos-routing-instance" {
```

```
          interface "$junos-interface-name";
            routing-options {
              access {
                route $junos-framed-route-ip-address-prefix {
                  next-hop $junos-framed-route-nexthop;
                  metric $junos-framed-route-cost;
                  preference $junos-framed-route-distance;
                }
              }
              access-internal {
                route $junos-subscriber-ip-address {
                  qualified-next-hop $junos-interface-name;
                }
              }
            }
          }
        }
      }
      interfaces "$junos-interface-ifd-name" {
        unit "$junos-interface-unit" {
          encapsulation multilink_ppp;
          mrru 1500;
          short-sequence;
          fragment-threshold 320;
          family inet
          }
        }
      }
    class-of-service {
        traffic-control-profiles {
          tcp2 {
            scheduler-map "$junos-cos-scheduler-map";
            shaping-rate "$junos-cos-shaping-rate";
            guaranteed-rate "$junos-cos-guaranteed-rate";
            delay-buffer-rate "$junos-cos-delay-buffer-rate";
          }
        }
        interfaces {
          "$junos-interface-ifd-name" {
            unit "$junos-interface-unit" {
              output-traffic-control-profile tcp2;
              fragmentation-map fragmap-2
              }
           }
        }
      }
```

Verification

Confirm that the configuration is working properly.

- Verifying the Subscriber Information on page 285
- Verifying Mixed Mode Support with a Dynamic MLPPP-Capable Subscriber on page 285
- Verifying Tunneled MLPPP Over LAC Interfaces on page 286

Verifying the Subscriber Information

Purpose Verify that the subscriber information for dynamic MLPPP over LNS is correct.

Action

```
root@haverhill> show subscribers extensive
Type: L2TP
User Name: lns-client
IP Address: 80.80.80.20
IP Netmask: 255.255.255.0
Logical System: default
Routing Instance: default
Interface: si-1/0/0.1073741824
Interface type: Dynamic
Dynamic Profile Name: ml-lns-member-prof
Dynamic Profile Version: 1
State: Active
Radius Accounting ID: 20
Session ID: 20
Bundle Session ID: 21
Login Time: 2011-04-11 10:55:13 PDT

Type: MLPPP
User Name: lns-client
IP Address: 80.80.80.20
IP Netmask: 255.255.255.0
Logical System: default
Routing Instance: default
Interface: si-3/0/0.1073741825
Interface type: Dynamic
Dynamic Profile Name: ml-bundle-prof
Dynamic Profile Version: 1
State: Active
Radius Accounting ID: 21
Session ID: 21
Underlying Session ID: 20
Login Time: 2011-04-11 07:55:59 PDT
```

Meaning Subscriber information for interface **si-1/0/0.1073741824** has been configured for MLPPP with interface type of dynamic.

Verifying Mixed Mode Support with a Dynamic MLPPP-Capable Subscriber

Purpose Verify that mixed mode interfaces negotiated correctly for the single link PPP using a dynamic MLPPP-capable subscriber.

Action

```
root@haverhill> show interfaces extensive pp0.1073741832
Logical interface pp0.1073741832 (Index 489) (SNMP ifIndex 712)
   (Generation 299)
    Flags: Up Point-To-Point SNMP-Traps 0x4000 Encapsulation: PPPoE
    PPPoE:
      State: SessionUp, Session ID: 40,
      Session AC name: haverhill1, Remote MAC address: 00:00:64:3c:01:02,
      Underlying interface: ge-1/0/0.44 (Index 376)
    Traffic statistics:
     Input  bytes  :                1213
     Output bytes  :                1672
     Input  packets:                  41
     Output packets:                  49
     IPv6 transit statistics:
      Input  bytes  :                  0
      Output bytes  :                  0
      Input  packets:                  0
      Output packets:                  0
    Local statistics:
     Input  bytes  :                 159
     Output bytes  :                1424
     Input  packets:                  10
     Output packets:                  18
    Transit statistics:
     Input  bytes  :                1054                    0 bps
     Output bytes  :                 248                    0 bps
     Input  packets:                  31                    0 pps
     Output packets:                  31                    0 pps
     IPv6 transit statistics:
      Input  bytes  :                  0
      Output bytes  :                  0
      Input  packets:                  0
      Output packets:                  0
  Keepalive settings: Interval 45 seconds, Up-count 1, Down-count 3
  LCP state: Opened
  NCP state: inet: Opened, inet6: Opened, iso: Not-configured, mpls:
  Not-configured
  CHAP state: Closed
  PAP state: Success
    Protocol inet, MTU: 65531, Generation: 384, Route table: 0
      Flags: Sendbcast-pkt-to-re
      Addresses, Flags: Is-Primary
        Destination: Unspecified, Local: 101.0.0.1, Broadcast: Unspecified,
        Generation: 297
    Protocol inet6, MTU: 65531, Generation: 385, Route table: 0
      Addresses, Flags: Is-Primary
        Destination: Unspecified, Local: 2030::1
    Generation: 298
        Destination: Unspecified, Local: fe80::2a0:a50f:fc64:6ef2
    Generation: 299
```

Meaning When a dynamic MLPPP-capable subscriber negotiates a single link PPP, the results are the same as a non-MLPPP subscriber; no bundle IFL or SDB session is created.

Verifying Tunneled MLPPP Over LAC Interfaces

Purpose Verify that the MLPPP over LAC member link IFL is correct.

Action

```
root@haverhill> show interfaces extensive pp0.1073756921
Logical interface pp0.1073756921 (Index 482) (SNMP ifIndex 706)
   (Generation 15542)
    Flags: Up Point-To-Point SNMP-Traps 0x4000 Encapsulation: PPPoE
    PPPoE:
      State: SessionUp, Session ID: 37,
      Session AC name: haverhill, Remote MAC address: 00:00:64:1d:01:02,
      Underlying interface: ge-1/0/0.2040 (Index 457)
    Traffic statistics:
     Input  bytes  :                 273
     Output bytes  :                 270
     Input  packets:                  13
     Output packets:                  10
    Local statistics:
     Input  bytes  :                 138
     Output bytes  :                 155
     Input  packets:                   6
     Output packets:                   3
    Transit statistics:
     Input  bytes  :                 135                    0 bps
     Output bytes  :                 115                    0 bps
     Input  packets:                   7                    0 pps
     Output packets:                   7                    0 pps
  Keepalive settings: Interval 45 seconds, Up-count 1, Down-count 3
  LCP state: Opened
  NCP state: inet: Not-configured, inet6: Not-configured, iso: Not-configured,
mpls:
  Not-configured
  CHAP state: Closed
  PAP state: Closed
    Protocol inet, MTU: 1492, Generation: 15534, Route table: 0
      Flags: Sendbcast-pkt-to-re
    Protocol mlppp, Multilink bundle: si-1/0/0.1073756922
    Service device pool: sipool-1, Dynamic profile: ml-bundle-prof
    MTU: 1526, Generation: 15533, Route table: 0
```

Meaning

When a PPPoE MLPPP session is tunneled, the bundle and member link binding remains. Although the bundle IFL does not participate in the control and forwarding path, it remains in the user-interface.

Related Documentation

- MLPPP Support for LNS and PPPoE Subscribers on MX Series Overview on page 217
- Mixed Mode Support for MLPPP and PPP Subscribers Overview on page 220
- Configuring L2TP Client Access to Support MLPPP for Dynamic Subscribers on page 241

Example: Configuring Dynamic PPPoE MLPPP Subscribers

This example shows how to configure dynamic Point-to-Point Protocol over Ethernet (PPPoE) multilink (MLPPP) subscribers.

- Requirements on page 288
- Overview on page 288

Requirements

This example uses the following hardware and software components:

- MX Series with MPC2s installed
- Junos OS Release 13.3 or later

Before you configure dynamic PPPoE MLPPP subscribers, be sure you have:

- If configuring a tunnel group using an inline service (**si**) interface, enabled the inline service (**si**) interface for PPPoE subscribers. See “Enabling Inline Service Interfaces for PPPoE and LNS Subscribers” on page 235.
- Configured the inline service (**si**) interface for PPPoE subscribers. See “Configuring Inline Service Interface for PPPoE and LNS Subscribers” on page 236.
- If configuring a tunnel group using a pool of service interfaces, configured service device pools for PPPoE subscribers. See “Configuring Service Device Pools for Load Balancing PPPoE and LNS Subscribers” on page 237.

Overview

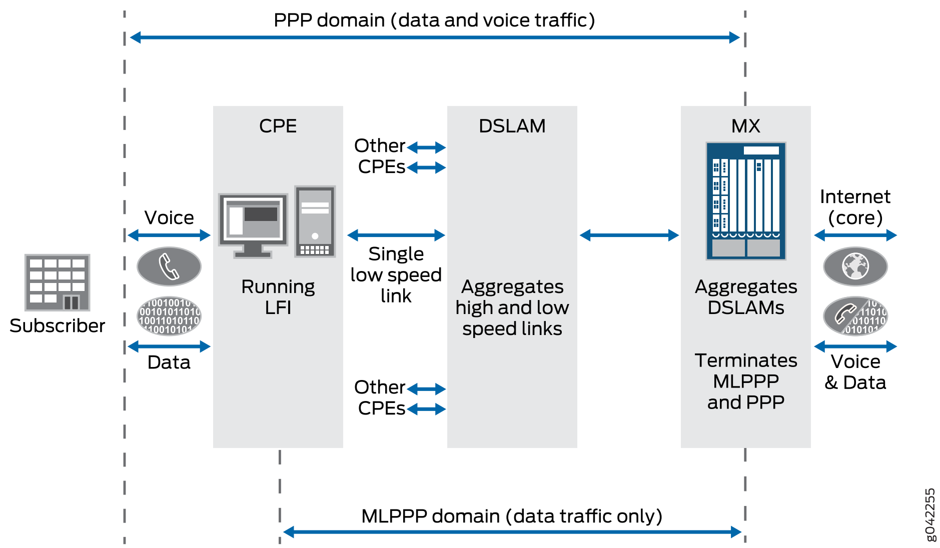
An MLPPP subscriber consists of two IFLs (logical interfaces), a member link, and a bundle. For dynamic PPPoE MLPPP subscribers, you configure the dynamic **pp0** member link IFLs using dynamic profiles. The **pp0** member link dynamic profile includes the **family mlppp** statement containing the dynamic profile name and the service interface (**si**), or a pool of service interfaces. This information is then used to create the dynamic bundle IFL.

Each dynamic bundle accepts only one dynamic member link. If more than one dynamic member link attempts to join the same dynamic bundle, the system fails the new member session.

Figure 19 on page 289 shows how the different types of traffic traverse through a network where the MX Series terminates PPPoE sessions.

Topology

Figure 19: PPP and MLPPP Traffic Terminated at MX Series



The following two domains are shown terminating traffic at the MX Series:

- PPP domain—Contains data and voice traffic
- MLPPP domain—Contains data traffic only

Configuration

To configure dynamic PPPoE MLPPP subscribers, perform these tasks:

- Configuring a Dynamic Profile for Dynamic pp0 Member Link IFL Without Mixed Mode Support on page 291
- Configuring a Dynamic Profile for Dynamic pp0 Member Link IFL With Mixed Mode Support on page 293
- Configuring a Dynamic Profile for the Dynamic Bundle IFL on page 296
- Results on page 298

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.

```
[edit]
set interfaces ge-1/0/0 flexible-vlan-tagging
set interfaces ge-1/0/0 unit 600 encapsulation ppp-over-ether vlan-id 600
set interfaces ge-1/0/0 unit 600 pppoe-underlying-options dynamic-profile
  ml-pp0-member-prof
set dynamic-profiles ml-pp0-member-prof
```

```
set dynamic-profiles ml-pp0-member-prof interfaces “$junos-interface-ifd-name” unit
  “$junos-interface-unit”
set dynamic-profiles ml-pp0-member-prof interfaces “$junos-interface-ifd-name” unit
  “$junos-interface-unit” pppoe-options underlying-interface
  "$junos-underlying-interface" server
set dynamic-profiles ml-pp0-member-prof interfaces “$junos-interface-ifd-name” unit
  “$junos-interface-unit” ppp-options pap chap lcp-restart-timer 5000
set dynamic-profiles ml-pp0-member-prof interfaces “$junos-interface-ifd-name” unit
  “$junos-interface-unit” family mlppp bundle “$junos-bundle-interface-name”
set dynamic-profiles ml-pp0-member-profinterfaces “$junos-interface-ifd-name” unit
  “$junos-interface-unit” family mlppp service-interface si-5/1/0
set dynamic-profiles ml-pp0-member-prof interfaces “$junos-interface-ifd-name” unit
  “$junos-interface-unit” family mlppp dynamic-profile ml-bundle-prof
set dynamic-profiles ml-pp0-member-prof interfaces “$junos-interface-ifd-name” unit
  “$junos-interface-unit” family inet

[edit]
set dynamic-profiles ml-bundle-prof
set dynamic-profiles ml-bundle-prof routing-instances "$junos-routing-instance" interface
  "$junos-interface-name"
set dynamic-profiles ml-bundle-prof routing-instances "$junos-routing-instance" interface
  "$junos-interface-name" routing-options access route
  $junos-framed-route-ip-address-prefix
set dynamic-profiles ml-bundle-prof routing-instances "$junos-routing-instance" interface
  "$junos-interface-name" routing-options access route
  $junos-framed-route-ip-address-prefix next-hop $junos-framed-route-nexthop
set dynamic-profiles ml-bundle-prof routing-instances "$junos-routing-instance" interface
  "$junos-interface-name" routing-options access route
  $junos-framed-route-ip-address-prefix metric $junos-framed-route-cost
set dynamic-profiles ml-bundle-prof routing-instances "$junos-routing-instance" interface
  "$junos-interface-name" routing-options access route
  $junos-framed-route-ip-address-prefix preference $junos-framed-route-distance
set dynamic-profiles ml-bundle-prof routing-instances "$junos-routing-instance" interface
  "$junos-interface-name" access-internal route $junos-subscriber-ip-address
set dynamic-profiles ml-bundle-prof routing-instances "$junos-routing-instance" interface
  "$junos-interface-name" access-internal route $junos-subscriber-ip-address
  qualified-next-hop $junos-interface-name
set dynamic-profiles ml-bundle-prof interfaces “$junos-interface-ifd-name” unit
  “$junos-interface-unit”
set dynamic-profiles ml-bundle-prof interfaces “$junos-interface-ifd-name” unit
  “$junos-interface-unit” encapsulation multilink-ppp
set dynamic-profiles ml-bundle-prof interfaces “$junos-interface-ifd-name” unit
  “$junos-interface-unit” mrru 1500
set dynamic-profiles ml-bundle-prof interfaces “$junos-interface-ifd-name” unit
  “$junos-interface-unit” short-sequence
set dynamic-profiles ml-bundle-prof interfaces “$junos-interface-ifd-name” unit
  “$junos-interface-unit” fragment-threshold 320
set dynamic-profiles ml-bundle-prof interfaces “$junos-interface-ifd-name” unit
  “$junos-interface-unit” family inet
set dynamic-profiles ml-bundle-prof class-of-service traffic-control-profiles tcp2
set dynamic-profiles ml-bundle-prof class-of-service traffic-control-profiles tcp2
  scheduler-map "$junos-cos-scheduler-map”
set dynamic-profiles ml-bundle-prof class-of-service traffic-control-profiles tcp2
  shaping-rate "$junos-cos-shaping-rate"
```


set dynamic-profiles ml-bundle-prof class-of-service traffic-control-profiles tcp2 guaranteed-rate "$junos-cos-guaranteed-rate"
set dynamic-profiles ml-bundle-prof class-of-service traffic-control-profiles tcp2 delay-buffer-rate "$junos-cos-delay-buffer-rate"
set dynamic-profiles ml-bundle-prof class-of-service interfaces "$junos-interface-ifd-name" unit "$junos-interface-unit"
set dynamic-profiles ml-bundle-prof class-of-service interfaces "$junos-interface-ifd-name" unit "$junos-interface-unit" output-traffic-control-profile tcp2
set dynamic-profiles ml-bundle-prof class-of-service interfaces "$junos-interface-ifd-name" unit "$junos-interface-unit" fragmentation-map fragmap-2

Configuring a Dynamic Profile for Dynamic pp0 Member Link IFL Without Mixed Mode Support

Step-by-Step Procedure

The following example requires that you navigate various levels in the configuration hierarchy.

You configure the dynamic **pp0** member link IFLs by using dynamic profiles and including the **family mlppp** statement. The **family mlppp** statement contains the **dynamic-profile *name***, and either the **service-interface** or the **service-device-pool** used to create the dynamic bundle IFL. If you configure a **service-device-pool**, an inline services (**si**) interface is selected from the pool to create the dynamic bundle IFL using a round-robin method.

You must also configure the **family inet** statement in the tunneled **pp0** member link dynamic profile. The **family inet** statement enables the L2TP long route to be installed and supported for the lookup engine to steer control packets to the Routing Engine.

NOTE: Optionally, you can configure the dynamic profile to support mixed mode to enable PPP subscribers to successfully log in using the dynamic profile. See "Configuring a Dynamic Profile for Dynamic pp0 Member Link IFL With Mixed Mode Support" on page 293 for the additional configuration commands required.

The following example shows how to configure dynamic **pp0** member link IFLs over flexible VLAN to support PPPoE MLPPP subscribers.

1. Create the Gigabit Ethernet underlying interface for the dynamic profile, **ge-1/0/0** and enable flexible VLAN tagging.

   [edit interfaces]
   user@host# **set ge-1/0/0 flexible vlan-tagging**

2. For the **ge-1/0/0** interface, configure PPP over Ethernet encapsulation for VLAN 600.

   [edit interfaces ge-1/0/0]
   user@host# **set unit 600 encapsulation ppp-over-ether vlan-id 600**

3. Configure the PPPoE underlying interface and set its dynamic profile.

   [edit interfaces ge-1/0/0 unit 600]

user@host# **set pppoe-underlying-options dynamic-profile ml-pp0-member-prof**

4. Specify the dynamic profile that you previously set as the PPPoE underlying interface dynamic profile.

   [edit dynamic-profiles]
   user@host# **set ml-pp0-member-prof**

5. Configure the interface for the dynamic profile by setting the predefined dynamic interface variable *$junos-interface-ifd-name*, and the logical interface unit by setting the predefined unit number variable *$junos-interface-unit*. The interface and unit number variables are dynamically replaced with the interface and unit number that the subscriber accesses when connecting to the MX Series.

   NOTE: The interface setting for a dynamic profile for PPPoE sessions can use either of the following code formats:

   - **set interfaces pp0**

     or

   - **set interfaces "$junos-interface-ifd-name"**

   This example uses **set interfaces "$junos-interface-ifd-name"**.

   [edit dynamic-profiles ml-pp0-member-prof]
   user@host# **set interfaces "$junos-interface-ifd-name" unit "$junos-interface-unit"**

6. For the *$junos-interface-ifd-name* interface, configure the underlying interface for the PPPoE options and PPPoE server mode for a dynamic PPPoE logical interface in a dynamic profile.

   [edit dynamic-profiles ml-pp0-member-prof interfaces "$junos-interface-ifd-name" unit "$junos-interface-unit"]
   user@host# **set pppoe-options underlying-interface "$junos-underlying-interface" server**

7. Configure PPP-specific interface properties in a dynamic profile: **pap**, **chap**, and set the **lcp-restart-timer** to 5000.

   [edit dynamic-profiles ml-pp0-member-prof interfaces "$junos-interface-ifd-name" unit "$junos-interface-unit"]
   user@host# **set ppp-options pap chap lcp-restart-timer 5000**

8. Enable MLPPP support for dynamic PPPoE MLPPP subscribers and configure the dynamic bundle interface (IFL) by setting the predefined dynamic bundle interface variable *$junos-bundle-interface-name*.

   NOTE: The **family mlppp** statement determines whether MLPPP is supported for subscribers coming in from the subscriber interface.

[edit dynamic-profiles ml-pp0-member-prof interfaces "$junos-interface-ifd-name" unit "$junos-interface-unit"]
user@host# **set family mlppp bundle "$junos-bundle-interface-name"**

9. Specify the service interface for the dynamic PPPoE MLPPP subscribers.

   [edit dynamic-profiles ml-pp0-member-prof interfaces "$junos-interface-ifd-name" unit "$junos-interface-unit" family mlppp]
   user@host# **set service-interface si-5/1/0**

10. Specify the dynamic profile name for the bundle.

    [edit dynamic-profiles ml-pp0-member-prof interfaces "$junos-interface-ifd-name" unit "$junos-interface-unit" family mlppp]
    user@host# **set dynamic-profile ml-bundle-prof**

11. Enable support for PPPoE tunneled subscribers and the LAC long route.

    [edit dynamic-profiles ml-lns-member-prof interfaces "$junos-interface-ifd-name" unit "$junos-interface-unit"]
    user@host# **set family inet**

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

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

Configuring a Dynamic Profile for Dynamic pp0 Member Link IFL With Mixed Mode Support

Step-by-Step Procedure

The following example requires that you navigate various levels in the configuration hierarchy.

Optionally, you can configure the dynamic profile to support mixed mode to enable PPP subscribers to successfully log in using the dynamic profile.

The following example shows the additional configurations required to support mixed mode for dynamic profiles.

NOTE: **The following configuration commands are not included in the** "CLI Quick Configuration" on page 289 **section.**

1. Configure dynamic **pp0** member link IFLs over flexible VLAN to support PPPoE MLPPP subscribers. See "Configuring a Dynamic Profile for Dynamic pp0 Member Link IFL Without Mixed Mode Support" on page 291, steps 1 through 4.

2. Specify the dynamic profile that you used to create the dynamic PPPoE MLPPP member link.

   [edit dynamic-profiles]
   user@host# **set ml-pp0-member-prof**

3. When the customer premises equipment (CPE) is for a dynamic virtual routing and forwarding (VRF) PPP subscriber, you must configure the routing instance and its interface.

   [edit dynamic-profiles ml-pp0-member-prof]
   user@host# **set routing-instances "$junos-routing-instance" interface "$junos-interface-name"**

4. Configure the access route for the routing options.

   [edit dynamic-profiles ml-pp0-member-prof routing-instances "$junos-routing-instance" interface "$junos-interface-name"]
   user@host# **set routing-options access route $junos-framed-route-ip-address-prefix**

5. Configure the next-hop, metric, and preference for the router.

   [edit dynamic-profiles ml-pp0-member-prof routing-instances "$junos-routing-instance" interface "$junos-interface-name" routing-options access route $junos-framed-route-ip-address-prefix]
   user@host# **set next-hop $junos-framed-route-nexthop**
   user@host# **set metric $junos-framed-route-cost**
   user@host# **set preference $junos-framed-route-distance**

6. Configure the internal access route for the routing options.

   [edit dynamic-profiles ml-pp0-member-prof routing-instances "$junos-routing-instance" interface "$junos-interface-name"]
   user@host# **set routing-options access-internal route $junos-subscriber-ip-address**

7. Configure the qualified next-hop for the internal route.

   [edit dynamic-profiles ml-pp0-member-prof routing-instances "$junos-routing-instance" interface "$junos-interface-name" routing-options access-internal route $junos-subscriber-ip-address]
   user@host# **set qualified-next-hop $junos-interface-name**

8. Configure the basic settings for the dynamic profile. See "Configuring a Dynamic Profile for Dynamic pp0 Member Link IFL Without Mixed Mode Support" on page 291, steps 5 through 11.

   NOTE: To enable mixed mode support, when the CPE is a PPP subscriber, you must also add an unnumbered address, and input and output filters to the **family inet** statement.

   [edit dynamic-profiles ml-pp0-member-prof interfaces "$junos-interface-ifd-name" unit "$junos-interface-unit"]
   user@host# **set family inet unnumbered-address $junos-loopback-interface**
   user@host# **set family inet filter input "$junos-input-filter" output "$junos-output-filter"**

9. When the CPE is a PPP subscriber, you must also configure class of service and define the traffic control profile.

[edit dynamic-profiles ml-pp0-member-prof class-of-service
user@host# **set traffic-control-profiles tc-profile**

10. For the traffic-control profile, define the following settings: scheduler map, shaping rate, overhead accounting, guaranteed rate, and delay buffer rate.

    [edit dynamic-profiles ml-pp0-member-prof class-of-service traffic-control-profiles tc-profile
    user@host# **set scheduler-map "$junos-cos-scheduler-map"**
    user@host# **set shaping-rate "$junos-cos-shaping-rate"**
    user@host# **set overhead-accounting "$junos-cos-shaping-mode" bytes "$junos-cos-byte-adjust"**
    user@host# **set guaranteed-rate "$junos-cos-guaranteed-rate"**
    user@host# **set delay-buffer-rate "$junos-cos-delay-buffer-rate"**

11. Configure the interface for the dynamic profile by setting the predefined dynamic interface variable *$junos-interface-ifd-name*, and the logical interface unit by setting the predefined unit number variable *$junos-interface-unit*.

    [edit dynamic-profiles ml-pp0-member-prof class-of-service]
    user@host# **set interfaces "$junos-interface-ifd-name" unit "$junos-interface-unit"**

12. For the dynamic profile interface, define the following settings: output traffic control profile, classifiers, and rewrite rules.

    [edit dynamic-profiles ml-pp0-member-prof class-of-service interfaces "$junos-interface-ifd-name" unit "$junos-interface-unit"]
    user@host# **set output-traffic-control-profile tc-profile**
    user@host# **set classifiers dscp GEN-CLASSIFIER-IN**
    user@host# **set rewrite-rules dscp GEN-RW-OUT-DSCP**

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

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

Configuring a Dynamic Profile for the Dynamic Bundle IFL

Step-by-Step Procedure

The following example requires that you navigate various levels in the configuration hierarchy.

To configure the dynamic profile for the dynamic bundle IFL, you specify the **encapsulation multilink-ppp** statement within the dynamic profile. The **dynamic profile** for the dynamic bundle IFL is referenced from the **dynamic profile** for dynamic PPPoE and LNS member link IFLs.

You must configure the **fragmentation-maps** statement statically using **class-of-service** and assign them in the bundle dynamic profile. You can also set these optional MLPPP parameters: MRRU, short sequence, and fragment-threshold. The following example shows how to configure the dynamic profile for the dynamic bundle IFL:

1. Specify the dynamic profile name for the bundle.

   [edit dynamic-profiles}
   user@host# **set ml-bundle-prof**

2. Although MLPPP member links process authentication and routing-instance assignments, if a non-default routing-instance is assigned, you must configure the bundle IFL under the assigned routing-instance. As a result, you must also configure routing-instances in the bundle dynamic-profile.

   [edit dynamic-profiles ml-bundle-prof]
   user@host# **set routing-instances "$junos-routing-instance" interface "$junos-interface-name"**

3. Configure the access route for the routing options.

   [edit dynamic-profiles ml-bundle-prof routing-instances "$junos-routing-instance" interface "$junos-interface-name"]
   user@host# **set routing-options access route $junos-framed-route-ip-address-prefix**

4. Configure the next-hop, metric, and preference for the router.

   [edit dynamic-profiles ml-bundle-prof routing-instances "$junos-routing-instance" interface "$junos-interface-name" routing-options access route $junos-framed-route-ip-address-prefix]
   user@host# **set next-hop $junos-framed-route-nexthop**
   user@host# **set metric $junos-framed-route-cost**
   user@host# **set preference $junos-framed-route-distance**

5. Configure the internal access route for the routing options.

   [edit dynamic-profiles ml-bundle-prof routing-instances "$junos-routing-instance" interface "$junos-interface-name"]
   user@host# **set routing-options access-internal route $junos-subscriber-ip-address**

6. Configure the qualified next-hop for the internal route.

   [edit dynamic-profiles ml-bundle-prof routing-instances "$junos-routing-instance" interface "$junos-interface-name" routing-options access-internal route $junos-subscriber-ip-address]

user@host# **set qualified-next-hop $junos-interface-name**

7. Configure the interface for the dynamic profile by setting the predefined dynamic interface variable *$junos-interface-ifd-name*, and the logical interface unit by setting the predefined unit number variable *$junos-interface-unit*. The interface and unit number variables are dynamically replaced with the interface and unit number that the subscriber accesses when connecting to the MX Series.

   [edit dynamic-profiles ml-bundle-prof]
   user@host# **set interfaces "$junos-interface-ifd-name" unit "$junos-interface-unit"**

8. Configure the **encapsulation multilink-ppp** statement to enable MLPPP bundling for the dynamic profile.

   [edit dynamic-profiles ml-bundle-prof interfaces "$junos-interface-ifd-name" unit "$junos-interface-unit"]
   user@host# **set encapsulation multilink-ppp**

9. Configure the following MLPPP options for this example:

   - **mrru**—Specifies the maximum received reconstructed unit value ranging from 1500 through 4500 bytes.
   - **fragment-threshold**—Applies to all packets and forwarding classes, ranging from 128 through 16,320 bytes.
   - **short-sequence**—Determines the header format for the MLPPP. Default is **long-sequence**.

   [edit dynamic-profiles ml-bundle-prof interfaces "$junos-interface-ifd-name" unit "$junos-interface-unit"]
   user@host# **set mrru 1500**
   user@host# **set fragment-threshold 320**
   user@host# **set short-sequence**

10. Enable support for MLPP subscribers.

   [edit dynamic-profiles ml-bundle-prof interfaces "$junos-interface-ifd-name" unit "$junos-interface-unit"]
   user@host# **set family inet**

11. To enable **fragmentation-maps** support, you must configure class of service and define the traffic control profile.

   [edit dynamic-profiles ml-bundle-prof class-of-service]
   user@host# **set traffic-control-profiles tcp2**

12. For the traffic-control profile, define the following settings: scheduler map, shaping rate, guaranteed rate, and delay buffer rate.

   [edit dynamic-profiles ml-bundle-prof class-of-service traffic-control-profiles tcp2]
   user@host# **set scheduler-map "$junos-cos-scheduler-map"**
   user@host# **set shaping-rate "$junos-cos-shaping-rate"**
   user@host# **set guaranteed-rate "$junos-cos-guaranteed-rate"**
   user@host# **set delay-buffer-rate "$junos-cos-delay-buffer-rate"**

13. Configure the underlying interface for the dynamic profile by setting the predefined dynamic interface variable *$junos-interface-ifd-name*, and the underlying logical interface unit by setting the predefined unit number variable *$junos-interface-unit*. The interface and unit number variables are dynamically replaced with the interface and unit number that the subscriber accesses when connecting to the MX Series.

    [edit dynamic-profiles ml-bundle-prof class-of-service]
    user@host# **set interfaces “$junos-interface-ifd-name” unit "$junos-interface-unit"**

14. For the dynamic profile interface, define the output traffic control profile.

    [edit dynamic-profiles ml-bundle-prof class-of-service interfaces “$junos-interface-ifd-name” unit "$junos-interface-unit"]
    user@host# **set output-traffic-control-profile tcp2**

15. Define the fragmentation-map required for dynamic profile bundles and used to enable link fragmentation and interleaving (LFI).

    [edit dynamic-profiles ml-bundle-prof class-of-service interfaces “$junos-interface-ifd-name” unit "$junos-interface-unit"]
    user@host# **set fragmentation-map fragmap-2**

16. 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 dynamic-profiles** command with the sub-hierarchy levels **interfaces**. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
user@host# show interfaces ge-1/0/0
  interfaces {
    ge-1/0/0 {
      flexible- vlan-tagging;
      unit 600 {
        encapsulation ppp-over-ether;
        vlan-id 600;
        pppoe-underlying-options {
          dynamic-profile ml-pp0-member-prot;
        }
      }
    }
  }
```

Dynamic profile for dynamic PPPoE member link IFL without mixed mode:

```
user@host# show dynamic-profiles mlp-pp0-member-profile
  dynamic-profile mlp-pp0-member-profile {
    interface $"$junos-interface-ifd-name" {
      unit "$junos-interface-unit" {
        pppoe-options {
          underlying-interface "$junos-underlying-interface";
```

```
            server;
          }
          ppp-options {
            pap;
            chap;
            lcp-restart-timer 5000;
          }
          family mlppp {
            bundle $junos-bundle-interface-name;
            service-interface si-5/1/0;
            dynamic-profile ml-bundle-prof;
          }
          family inet
        }
      }
    }
```

Dynamic profile for dynamic PPPoE member link IFL with mixed mode:

```
user@host# show dynamic-profiles mlp-pp0-member-profile
  dynamic-profile ml-pp0-member-prof {
    routing-instances {
      "$junos-routing-instance" {
        interface "$junos-interface-name";
          routing-options {
            access {
              route $junos-framed-route-ip-address-prefix {
                next-hop $junos-framed-route-nexthop;
                metric $junos-framed-route-cost;
                preference $junos-framed-route-distance;
              }
            }
            access-internal {
              route $junos-subscriber-ip-address {
                qualified-next-hop $junos-interface-name;
              }
            }
          }
        }
      }
    }
    interfaces $"$junos-interface-ifd-name" {
      unit "$junos-interface-unit" {
        pppoe-options {
          underlying-interface "$junos-underlying-interface";
          server;
        }
        ppp-options {
          pap;
          chap;
          lcp-restart-timer 5000;
        }
        family mlppp {
          bundle $junos-bundle-interface-name;
          service-interface si-5/1/0;
          dynamic-profile ml-bundle-prof;
```

```
            }
            family inet {
              unnumbered-address $junos-loopback-interface;
              filter {
                input "$junos-input-filter";
                output "$junos-output-filter";
              }
            }
          }
      class-of-service {
        traffic-control-profiles {
            tc-profile {
              scheduler-map "$junos-cos-scheduler-map";
              shaping-rate "$junos-cos-shaping-rate";
            overhead-accounting "$junos-cos-shaping-mode" bytes "$junos-cos-byte-adjust";

              guaranteed-rate "$junos-cos-guaranteed-rate";
              delay-buffer-rate "$junos-cos-delay-buffer-rate";
            }
          }
          interfaces {
            "$junos-interface-ifd-name" {
              unit "$junos-interface-unit" {
                output-traffic-control-profile tc-profile;
                classifiers {
                  dscp GEN-CLASSIFIER-IN;
                }
                rewrite-rules {
                  dscp GEN-RW-OUT-DSCP;
                }
              }
            }
          }
        }
      }
```

Verification

Confirm that the configuration is working properly.

Verifying the Subscriber Information

Purpose Verify that the subscriber information for dynamic MLPPP over PPPoE is correct.

Action

```
root@haverhill> show subscribers extensive
Type: PPPoE
User Name: dual-stack-v4v6-nas@almach.com
Logical System: default
Routing Instance: default
Interface: pp0.1073741824
Interface type: Dynamic
Underlying Interface: ge-1/1/0.3000
Dynamic Profile Name: DS-lac-mlppp-link-ipv6
MAC Address: 30:00:64:30:00:02
State: Active
PPP State: Tunneled
Local IP Address: 210.1.1.1
Remote IP Address: 210.1.1.2
Radius Accounting ID: 5
Session ID: 5
Bundle Session ID: 6
VLAN Id: 3000
Login Time: 2013-03-28 15:42:30 PDT

Type: MLPPP
Logical System: default
Routing Instance: default
Interface: si-1/1/0.1073741825
Interface type: Dynamic
Underlying Interface: si-1/1/0.1073741825
Dynamic Profile Name: DS-mlppp-bundle-ipv6
State: Active
PPP State: Tunneled
Local IP Address: N/A
Remote IP Address: N/A
Radius Accounting ID: 6
Session ID: 6
Underlying Session ID: 5
Login Time: 2013-03-28 15:42:30 PDT
```

Meaning When a PPPoE MLPPP session is tunneled, the bundle and member link binding is maintained. The **PPP State** setting for both bundle and member link is set to **Tunneled**. Although there is no NCP negotiation over the bundle, the bundle session remains active.

Verifying Mixed Mode Support with a Dynamic MLPPP-Capable Subscriber

Purpose Verify that mixed-mode interfaces negotiated correctly for the single link PPP using a dynamic MLPPP-capable subscriber.

Action

```
root@haverhill> show interfaces extensive pp0.1073741832
Logical interface pp0.1073741832 (Index 489) (SNMP ifIndex 712)
   (Generation 299)
    Flags: Up Point-To-Point SNMP-Traps 0x4000 Encapsulation: PPPoE
    PPPoE:
      State: SessionUp, Session ID: 40,
      Session AC name: haverhill1, Remote MAC address: 00:00:64:3c:01:02,
      Underlying interface: ge-1/0/0.44 (Index 376)
    Traffic statistics:
     Input  bytes  :               1213
     Output bytes  :               1672
     Input  packets:                 41
     Output packets:                 49
     IPv6 transit statistics:
      Input  bytes  :                 0
      Output bytes  :                 0
      Input  packets:                 0
      Output packets:                 0
    Local statistics:
     Input  bytes  :                159
     Output bytes  :               1424
     Input  packets:                 10
     Output packets:                 18
    Transit statistics:
     Input  bytes  :               1054                    0 bps
     Output bytes  :                248                    0 bps
     Input  packets:                 31                    0 pps
     Output packets:                 31                    0 pps
     IPv6 transit statistics:
      Input  bytes  :                 0
      Output bytes  :                 0
      Input  packets:                 0
      Output packets:                 0
  Keepalive settings: Interval 45 seconds, Up-count 1, Down-count 3
  LCP state: Opened
  NCP state: inet: Opened, inet6: Opened, iso: Not-configured, mpls:
  Not-configured
  CHAP state: Closed
  PAP state: Success
    Protocol inet, MTU: 65531, Generation: 384, Route table: 0
      Flags: Sendbcast-pkt-to-re
      Addresses, Flags: Is-Primary
        Destination: Unspecified, Local: 101.0.0.1, Broadcast: Unspecified,
        Generation: 297
    Protocol inet6, MTU: 65531, Generation: 385, Route table: 0
      Addresses, Flags: Is-Primary
        Destination: Unspecified, Local: 2030::1
    Generation: 298
        Destination: Unspecified, Local: fe80::2a0:a50f:fc64:6ef2
    Generation: 299
```

Meaning When a dynamic MLPPP-capable subscriber negotiates a single link PPP, the results are the same as a non-MLPPP subscriber; no bundle IFL or SDB session is created.

Verifying Tunneled PPPoE MLPPP Interfaces

Purpose Verify that the PPPoE MLPPP member link IFL is correct.

Action

```
root@haverhill> show interfaces extensive pp0.1073756921
Logical interface pp0.1073756921 (Index 482) (SNMP ifIndex 706)
   (Generation 15542)
    Flags: Up Point-To-Point SNMP-Traps 0x4000 Encapsulation: PPPoE
    PPPoE:
      State: SessionUp, Session ID: 37,
      Session AC name: haverhill, Remote MAC address: 00:00:64:1d:01:02,
      Underlying interface: ge-1/0/0.2040 (Index 457)
    Traffic statistics:
     Input  bytes  :                 273
     Output bytes  :                 270
     Input  packets:                  13
     Output packets:                  10
    Local statistics:
     Input  bytes  :                 138
     Output bytes  :                 155
     Input  packets:                   6
     Output packets:                   3
    Transit statistics:
     Input  bytes  :                 135                    0 bps
     Output bytes  :                 115                    0 bps
     Input  packets:                   7                    0 pps
     Output packets:                   7                    0 pps
  Keepalive settings: Interval 45 seconds, Up-count 1, Down-count 3
  LCP state: Opened
  NCP state: inet: Not-configured, inet6: Not-configured, iso: Not-configured,
mpls:
  Not-configured
  CHAP state: Closed
  PAP state: Closed
    Protocol inet, MTU: 1492, Generation: 15534, Route table: 0
      Flags: Sendbcast-pkt-to-re
    Protocol mlppp, Multilink bundle: si-1/0/0.1073756922
    Service device pool: sipool-1, Dynamic profile: ml-bundle-prof
    MTU: 1526, Generation: 15533, Route table: 0
```

Meaning

When a PPPoE MLPPP session is tunneled, the bundle and member link binding remains. Although the bundle IFL does not participate in the control and forwarding path, it remains in the user interface.

Related Documentation

- MLPPP Support for LNS and PPPoE Subscribers on MX Series Overview on page 217
- Mixed Mode Support for MLPPP and PPP Subscribers Overview on page 220
- MLPPP Bundles and Inline Service Logical Interfaces Overview on page 233

CHAPTER 28

Configuring Dynamic PPP Subscriber Services

Dynamic PPP Subscriber Services for Static MLPPP Interfaces Overview

Dynamic subscriber services are supported for MLPPP bundle interfaces, with certain interface and hardware restrictions. See “Hardware Requirements for PPP Subscriber Services on Non-Ethernet Interfaces” on page 306. Multiclass MLPPP enables the relative prioritization of up to eight classes of traffic over an MLPPP bundle, but only on link services intelligent queuing (IQ) (LSQ) interfaces.

RADIUS previously supported only authentication for MLPPP. Address management, service deactivation, and dynamic selection of subscriber properties based on RADIUS user ID are now also supported.

RADIUS can dynamically allocate IPv4 addresses for MLPPP connections. When the first subscriber logs in, an address is allocated. The same address is allocated to all links in a bundle. Any other address provided for any of the links is ignored. The IP address is released for re-allocation when the last member link in a bundle logs out. Similar to the address allocation, the services configured for the first subscriber to log in are configured for all subsequent subscribers in the bundle.

The Acct-Multi-Session-Id [50] attribute enables RADIUS to link multiple related sessions into a single log file. RADIUS uses the session database (SDB) bundle session ID for the value of Acct-Multi-Session-Id. This bundle ID enables RADIUS to initiate a disconnect for an entire bundle. By tracking the member link sessions, RADIUS is also able to disconnect the individual member links in a bundle.

The Acct-Link-Count [51] attribute records the number of links present in a multilink session at the time the accounting record is generated.

Related Documentation

- Hardware Requirements for PPP Subscriber Services on Non-Ethernet Interfaces on page 306
- Configuring PPP Subscriber Services for MLPPP Bundles on page 306

Hardware Requirements for PPP Subscriber Services on Non-Ethernet Interfaces

PPP subscriber services are supported for MLPPP bundle interfaces. These services require the following hardware:

- M120 router or M320 router
- Channelized DS3/E3 Enhanced IP PIC (PB-4CHDS3-E3-IQE-BNC) to support MLPPP subscriber access
- An Adaptive Services PIC or Multiservices PIC to support subscriber services on LSQ MLPPP bundle interfaces

Subscriber services are not supported for single-link PPP interfaces with this hardware.

Related Documentation

- Dynamic PPP Subscriber Services for Static MLPPP Interfaces Overview on page 305

Configuring PPP Subscriber Services for MLPPP Bundles

You can configure PPP subscriber services for static LSQ MLPPP bundle interfaces.

To configure PPP subscriber services for static LSQ MLPPP bundle interfaces:

1. Enable PPP subscriber services for the interfaces.

   See “Enabling PPP Subscriber Services for Static Non-Ethernet Interfaces” on page 307.
2. Attach a dynamic profile to the MLPPP bundle interface.

   See “Attaching Dynamic Profiles to MLPPP Bundles” on page 307.

Related Documentation

- Hardware Requirements for PPP Subscriber Services on Non-Ethernet Interfaces on page 306
- Example: Minimum MLPPP Dynamic Profile on page 308
- Example: Configuring CoS on Static LSQ MLPPP Bundle Interfaces on page 308

Enabling PPP Subscriber Services for Static Non-Ethernet Interfaces

You can enable PPP subscriber services for certain non-Ethernet interface types on particular associated PICs. Supported interfaces are listed in "Hardware Requirements for PPP Subscriber Services on Non-Ethernet Interfaces" on page 306.

To enable PPP subscriber services on supported non-Ethernet interfaces:

- Configure PPP subscriber services.

```
[edit chassis]
user@host# set ppp-subscriber-services enable
```

To disable PPP subscriber services on supported non-Ethernet interfaces:

- Disable PPP subscriber services.

```
[edit chassis]
user@host# set ppp-subscriber-services disable
```

Related Documentation

- For hardware requirements, see Hardware Requirements for PPP Subscriber Services on Non-Ethernet Interfaces on page 306
- Configuring PPP Subscriber Services for MLPPP Bundles on page 306

Attaching Dynamic Profiles to MLPPP Bundles

You can attach a dynamic profile to a static MLPPP bundle interface. When a PPP subscriber logs in on a member link, the specified dynamic profile is instantiated and the services defined in the profile are applied to the LSQ bundle interface.

To attach a dynamic profile to a static LSQ MLPPP bundle interface:

1. Specify that you want to configure PPP options.

```
[edit interfaces lsq-3/3/0 unit 0]
user@host# edit ppp-options
```

2. Specify the dynamic profile you want to associate with the interface.

```
[edit interfaces lsq-3/3/0 unit 0 ppp-options]
user@host# set dynamic-profile vod-profile-50
```

Related Documentation

- Hardware Requirements for PPP Subscriber Services on Non-Ethernet Interfaces on page 306
- Configuring PPP Subscriber Services for MLPPP Bundles on page 306

- *Dynamic Profiles Overview*

- Configuring PPP Subscriber Services for MLPPP Bundles on page 306
- Example: Minimum MLPPP Dynamic Profile on page 308
- Example: Configuring CoS on Static LSQ MLPPP Bundle Interfaces on page 308

Example: Minimum MLPPP Dynamic Profile

This example shows the minimum configuration for a dynamic profile that is used for static LSQ MLPPP bundle interfaces.

```
dynamic-profiles {
  mlppp-profile-1 {
    interfaces {
      "$junos-interface-ifd-name" {
        unit "$junos-underlying-interface-unit";
      }
    }
  }
}
```

Related Documentation

- Attaching Dynamic Profiles to MLPPP Bundles on page 307

Example: Configuring CoS on Static LSQ MLPPP Bundle Interfaces

This example shows how to configure dynamic subscriber services on MLPPP bundle interfaces. The MLPPP bundles must be configured on link services intelligent queuing (IQ) (LSQ) interfaces. The MLPPP interfaces must be statically configured.

To configure dynamic subscriber services on static LSQ MLPPP bundle interfaces:

1. Configure class of service features for the LSQ interfaces.

```
[edit]
class-of-service
  classifiers {
    inet-precedence inet_classifier {
      forwarding-class best-effort {
        loss-priority low code-points 000;
      }
      forwarding-class expedited-forwarding {
        loss-priority low code-points 011;
      }
      forwarding-class assured-forwarding {
        loss-priority low code-points 100;
      }
    }
  }
  fragmentation-maps {
    sample-fragmap {
      forwarding-class {
        best-effort {
          fragment-threshold 1000;
          multilink-class 1:
        }
        assured-forwarding {
          fragment-threshold 1000;
          multilink-class 2:
        }
```



```
            expedited-forwarding {
              multilink-class 3:
            }
          }
        }
      }
      forwarding-classes {
        queue 0 best-effort;
        queue 1 expedited-forwarding;
        queue 2 assured-forwarding;
      }
      # traffic classifiers are statically defined
        network traffic interface{
          classifiers {
            inet-precedence inet_classifier;
          }
        }
      scheduler-maps {
        allthree {
          forwarding-class best-effort scheduler be-scheduler;
          forwarding-class expedited-forwarding scheduler hiprior-sched;
          forwarding-class assured-forwarding scheduler vpn-sched;
        }
      }
      schedulers {
        be-scheduler {
          transmit-rate percent 30;
          priority low;
        }
        hiprior-scheduler {
          transmit-rate percent 40;
          priority strict-high;
        }
        vpn-sched {
          transmit-rate percent 30;
          medium-high;
        }
      }
    }
```

2. Configure the MLPPP bundle interfaces and the LSQ interfaces.

```
    [edit interfaces]
    t1-3/1/0:1:1 {
      keepalives interval 600;
      encapsulation ppp;
      unit 0 {
        ppp-options {
          lcp-restart-timer 5000;
        }
        family mlppp {
          bundle lsq-3/3/0.0;
        }
      }
    }
    t1-3/1/0:1:2 {
```

```
        keepalives interval 600;
        encapsulation ppp;
        unit 0 {
            ppp-options {
                lcp-restart-timer 5000;
            }
            family mlppp {
                bundle lsq-3/3/0.0;
            }
        }
    }
    lsq-3/3/0 {
        unit 0 {
            encapsulation multilink-ppp;
            multilink-max-classes 4;
            ppp-options {
                ncp-restart-timer 10000;
                dynamic-profile mlppp-profile;
            }
            family inet {
                address 192.168.1.1/32 {
                    destination 192.168.25.45;
                }
            }
        }
    }
```

3. Configure the dynamic profile that is applied to the MLPPP bundle interfaces.

```
[edit]
dynamic-profiles {
    mlppp-profile {
        interfaces {
            "$junos-interface-ifd-name" {
                unit junos-underlying-interface-unit" {
                    family inet {
                        filter {
                            input "$junos-input-filter";
                            output "$junos-output-filter";
                        }
                    }
                }
            }
        }
        class-of-service {
            interfaces {
                "$junos-interface-ifd-name" {
                    unit junos-underlying-interface-unit" {
                        output-traffic-control-profile tcp1;
                        fragmentation-map sample-fragmap;
                    }
                }
            }
            traffic-control-profiles {
                tcp1 {
                    scheduler-map "junos-cos-scheduler-map";
```



```
            shaping-rate "$junos-cos-shaping-rate";
            guaranteed-rate "$junos-cos-guaranteed-rate";
            delay-buffer-rate "$junos-cos-delay-buffer-rate";
          }
        }
        scheduler-maps {
          data_smap {
            forwarding-class be scheduler data_sch;
          }
        }
        schedulers {
          be_sch {
            ...
          }
        }
      }
    }
  }
```

Related Documentation

- Hardware Requirements for PPP Subscriber Services on Non-Ethernet Interfaces on page 306
- *Layer 2 Service Package Capabilities and Interfaces*

CHAPTER 29

Monitoring and Managing MLPPP for Subscriber Access

MLPPP Subscriber Accounting Statistics Overview

For broadband subscriber management edge router Point-to-Point Protocol (PPP) subscribers, the accounting statistics contain two groups:

- The aggregate (IPv4 and IPv6) statistics group consists of statistics reported through these RADIUS attributes: **Acct-Input-Octets**, **Acct-Output-Octets**, **Acct-Input-Packets**, and **Acct-Output-Packets**.
- The IPv6 portion of the aggregate statistics group reported through the Juniper Networks **ERX-VSA**s **151** through **156**.

Broadband subscriber management edge router PPP logical interfaces (IFLs) support accurate accounting statistics by excluding PPP control traffic, and incrementing packet and octets at the point where the packet is about to leave the router. The packet is not dropped by CoS, filters, or policers.

For MLPPP subscribers, accounting is performed for each member link (currently limited to one) and not the bundle. The bundle IFL supports accurate accounting statistics only, and the member link supports transit statistics only. As a result, the following restrictions apply for member link final aggregate statistics:

- Only aggregate statistics are available with no IPv6 specific statistics; for example, **ERX-VSA 151** to **156** are all zeros.
- Packets sent and received over the member link include fragments and non-fragmented packets.
- Octets sent and received are bytes in the fragments and non-fragmented packets.
- Aggregate statistics include packets that can be dropped in the router, such as CoS, filters, and policers.
- Aggregate statistics include PPP control packets (LCP, PAP, CHAP, and NCP) and keepalive packets.

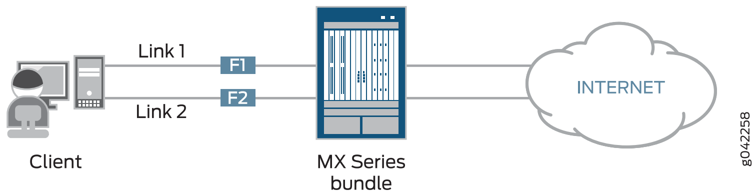
The following topics describe the statistics collection process in the lookup engine for member links and its bundle.

Member Link and Bundle Statistics Collection

MLPPP with MPC2 currently supports only one member link per bundle. However, support for accounting statistics must consider a true multilink scenario where multiple member links exist per bundle. From the lookup engine, only the bundle has the ability to maintain Layer 3 statistics. For an individual member link, only protocol-agnostic fragments (plus non-fragmented packets) are counted.

Figure 20 on page 315 shows an MLPPP client with two active member links and the statistics maintained by the lookup engine. For MLPPP with MPC2, each member link and bundle can reside on different lookup engines from where the accounting statistics are maintained.

Figure 20: MLPPP Client with Two Active Member Links



Client-to-Internet Traffic Statistics

When the client sends IP packets towards the Internet, they may be fragmented. For example, packet P1 is fragmented into F1 and F2, and the fragments belonging to a single packet can be sent on different links (Figure 20 on page 315).

- F1 is sent on Link 1
- F2 is sent on Link 2

When Link 1 on the MX Series receives fragment F1, it is identified as an MLPPP encapsulated fragment. Because IPv4 or IPv6 families are indicated on the first fragment, all of the incoming fragments are counted using a protocol-agnostic method before the fragment is forwarded to the bundle for reassembly.

- The protocol-agnostic incoming packet count is incremented by 1.
- The protocol-agnostic incoming byte count is incremented by the size of the fragment.

Similarly on Link 2, fragment F2 is also counted using a protocol-agnostic method, and then forwarded to the bundle for reassembly.

Fragment F1 arrives at the bundle and is stored along with its MLPPP header containing the sequence number with the **begin flag** set to 0, and the **end flag** set to 1.

Fragment F2 arrives at the bundle and is stored along with its MLPPP header containing the sequence number with the **begin flag** set to 1, and the **end flag set** to 0.

The pattern of monotonically increasing sequence numbers, **begin flag** set to 1 and **end flag** set to 1, causes fragments F1 and F2 to be reassembled into a single packet.

After the packet has been reassembled, the packet's Layer 3 type (either IPv4 or IPv6) is determined at the bundle. Then, the packets and bytes are counted according to its Layer 3 type at the bundle based on accurate accounting statistics:

- *bundleA_ipv4_packets_from_client* += 1
- *bundleA_ipv4_bytes_from_client* += *packet_size*

Or

- *bundleA_ipv6_packets_from_client* += 1

- *bundleA_ipv6_bytes_from_client* += *packet_size*

Internet-to-Client Traffic Statistics

In the reverse direction, Layer 3 packets come from the Internet to the bundle.

The packets and bytes are counted according to its Layer 3 type at the bundle:

- *bundleA_ipv4_packets_to_client* += 1
- *bundleA_ipv4_bytes_to_client* += *packet_size*

Or

- *bundleA_ipv6_packets_to_client* += 1
- *bundleA_ipv6_bytes_to_client* += *packet_size*

If the packets are fragmented, the fragments belonging to the same packet can be sent out different links. Because no IPv4 or IPv6 families are indicated on the links, all of the outgoing fragments are counted using a protocol-agnostic method.

- The protocol-agnostic outgoing packet count is incremented by 1.
- The protocol-agnostic outgoing byte count is incremented by the size of the fragment.

RADIUS Final Statistics Output Example

The following output example shows RADIUS final statistics:

```
User-Name = "mlUser1@isp1.com"
    Acct-Status-Type = Stop
    Acct-Session-Id = "786"
    Acct-Multi-Session-Id = "787"
    Acct-Input-Octets = 1068151928
    Acct-Output-Octets = 4268692096
    Acct-Session-Time = 61965
    Acct-Input-Packets = 406636696
    Acct-Output-Packets = 357477811
    Acct-Terminate-Cause = Lost-Carrier
    Service-Type = Framed-User
    Framed-Protocol = PPP
    Framed-IPv6-Pool = "v6-pool-21"
    Acct-Authentic = RADIUS
    Acct-Delay-Time = 0
    ERX-Dhcp-Mac-Addr = "0090.1a41.ec2d"
    Event-Timestamp = "Oct 19 2012 10:31:03 IST"
    Framed-IP-Address = 21.0.0.3
    Framed-IP-Netmask = 255.0.0.0
    ERX-Input-Gigapkts = 0
    Acct-Input-Gigawords = 6
    NAS-Identifier = "kalka"
    NAS-Port = 306184213
    NAS-Port-Id = "ge-1/1/9.21:21"
    NAS-Port-Type = Ethernet
    ERX-Output-Gigapkts = 0
```



```
Acct-Output-Gigawords = 4
ERX-Attr-151 = 0x00000000
ERX-Attr-152 = 0x00000000
ERX-Attr-153 = 0x00000000
ERX-Attr-154 = 0x00000000
ERX-Attr-155 = 0x00000000
ERX-Attr-156 = 0x00000000
NAS-IP-Address = 111.1.1.2
Acct-Unique-Session-Id = "03eeef735aef3520"
Timestamp = 1350604541
Request-Authenticator = Verified
```

Related Documentation

- MLPPP Bundles and Inline Service Logical Interfaces Overview on page 233
- MLPPP Support for LNS and PPPoE Subscribers on MX Series Overview on page 217
- Supported Features for MLPPP LNS and PPPoE Subscribers on the MX Series on page 219

PART 5

Configuring MPLS Pseudowire for Subscribers

CHAPTER 30

Configuring MPLS Pseudowire Subscriber Logical Interfaces

Pseudowire Subscriber Logical Interfaces Overview

Subscriber management supports the creation of subscriber interfaces over point-to-point MPLS pseudowires. The pseudowire subscriber interface capability enables service providers to extend an MPLS domain from the access-aggregation network to the service edge, where subscriber management is performed. Service providers can take advantage of MPLS capabilities such as failover, rerouting, and uniform MPLS label provisioning, while using a single pseudowire to service a large number of DHCP and PPPoE subscribers in the service network.

NOTE: Pseudowire subscriber logical interfaces are supported on Modular Port Concentrators (MPCs) with Ethernet Modular Interface Cards (MICs) only.

The pseudowire is a tunnel that is either an MPLS-based Layer 2 VPN or Layer 2 circuit. The pseudowire tunnel transports Ethernet encapsulated traffic from an access node (for example, a DSLAM or other aggregation device) to the MX Series router that hosts the subscriber management services. The termination of the pseudowire tunnel on the MX Series router is similar to a physical Ethernet termination, and is the point at which subscriber management functions are performed. A service provider can configure multiple pseudowires on a per-DSLAM basis and then provision support for a large number of subscribers on a specific pseudowire. Figure 21 on page 322 shows an MPLS network that provides subscriber management support.

At the access node end of the pseudowire, the subscriber traffic can be groomed into the pseudowire in a variety of ways, limited only by the number and types of interfaces that can be stacked on the pseudowire. You specify an anchor point, which identifies the logical tunnel interface that terminates the pseudowire tunnel at the access node.

Figure 21: MPLS Access Network with Subscriber Management Support

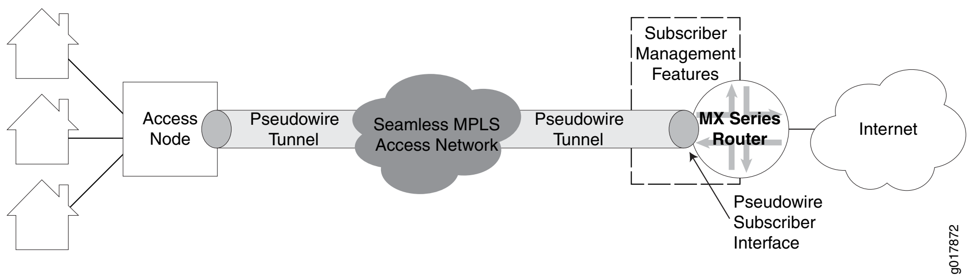
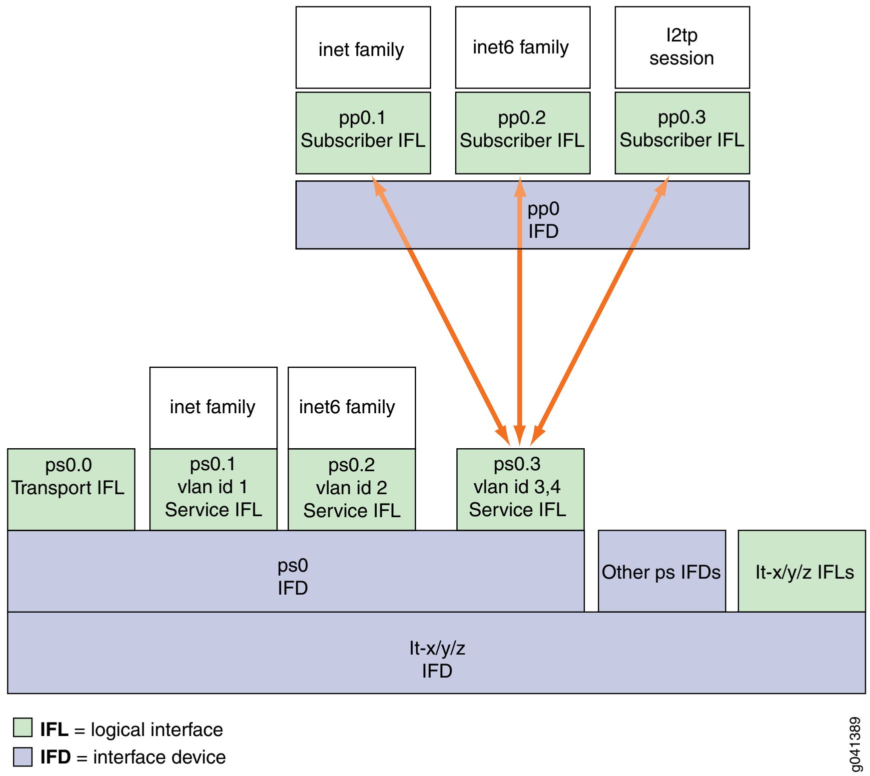


Figure 22 on page 323 shows the protocol stack for a pseudowire subscriber logical interface. The pseudowire is a virtual device that is stacked above the logical tunnel anchor point on the physical interface (the IFD), and supports a circuit-oriented Layer 2 protocol (either Layer 2 VPN or Layer 2 circuit). The Layer 2 protocol provides the transport and service logical interfaces, and supports the protocol family (IPv4, IPv6, or PPPoE).

Figure 22: Pseudowire Subscriber Interface Protocol Stack



The pseudowire configuration is transparent to the subscriber management applications and has no impact on the packet payloads that are used for subscriber management. Subscriber applications such as DHCP and PPPoE can be stacked over Layer 2 similar to the way in which they are stacked over a physical interface.

Related Documentation

- Configuring a Pseudowire Subscriber Logical Interface on page 323
- Hierarchical CoS on MPLS Pseudowire Subscriber Interfaces Overview on page 333

Configuring a Pseudowire Subscriber Logical Interface

A pseudowire subscriber logical interface terminates an MPLS pseudowire tunnel from an access node to the MX Series router that hosts subscriber management, and enables you to perform subscriber management services at the interface.

To create a pseudowire subscriber logical interface:

1. Specify the number of pseudowire logical interfaces that the router can support.

   See “Configuring the Maximum Number of Pseudowire Logical Interface Devices Supported on the Router” on page 325.

2. Configure the pseudowire subscriber logical interface device.

   See “Configuring a Pseudowire Subscriber Logical Interface Device” on page 325.

3. Configure the transport logical interface.

   See “Configuring the Transport Logical Interface for a Pseudowire Subscriber Logical Interface” on page 327.

4. Configure the signaling for the pseudowire subscriber interface. You can use either Layer 2 circuit signaling or Layer 2 VPN signaling. The two signaling types are mutually exclusive for a given pseudowire.

   - To configure Layer 2 circuit signaling, see “Configuring Layer 2 Circuit Signaling for Pseudowire Subscriber Logical Interfaces” on page 328.
   - To configure Layer 2 VPN signaling, see “Configuring Layer 2 VPN Signaling for Pseudowire Subscriber Logical Interfaces” on page 328.

5. Configure the service logical interface.

   See “Configuring the Service Logical Interface for a Pseudowire Subscriber Logical Interface” on page 330.

6. Configure the underlying interface device.

   See “Configuring an Underlying Interface for Dynamic PPPoE Subscriber Interfaces” on page 156.

7. Configure CoS parameters and BA classification.

   See “CoS Configuration Overview for MPLS Pseudowire Subscriber Interfaces” on page 334.

8. (Optional) Associate a dynamic profile with the pseudowire subscriber logical interface.

   You can associate DHCP, PPPoE, IP demux, and VLAN dynamic profiles with pseudowire subscriber logical interfaces. The support is similar to the typical Ethernet interface support.

   NOTE: When using a PPPoE dynamic profile to create a pseudowire subscriber logical interface over a demux interface device, the dynamic profile must explicitly specify the correct pseudowire interface device over which the interface is created. The dynamic profile does not automatically create the interface over the demux0 interface device, as is the case with a VLAN demux interface.

   For additional information about associating dynamic profiles to interfaces, see “Dynamic Profile Attachment to DHCP Subscriber Interfaces Overview” on page 113 and *Configuring VLAN Interfaces to Use Dynamic Profiles*.

9. (Optional) Configure interface set support for pseudowire subscriber logical interfaces.

   See *Configuring Interface Sets* and *Applying Interface Sets*.

10. (Optional) Stack PPPoE logical interfaces over a pseudowire logical device.

Related Documentation

- Pseudowire Subscriber Logical Interfaces Overview on page 321

Configuring the Maximum Number of Pseudowire Logical Interface Devices Supported on the Router

You must set the maximum number of pseudowire logical interface devices (pseudowire tunnels) that the router can use for subscriber logical interfaces. You can specify a maximum of 2048 pseudowire logical interface devices for an MX Series router. Each pseudowire device supports a maximum of 4000 subscriber logical interfaces.

A PFE can host a maximum of 2048 pseudowire logical interface devices, which is the chassis maximum. This PFE hosting support provides the configuration flexibility needed for special cases that might occur for business edge scenarios. However, you can exceed the available PFE resources as you configure additional services on the pseudowire logical interface devices ports. To support a scaled configuration, ensure that you populate the appropriate number of PFEs for the chassis, and that you distribute the pseudowire logical interface devices across the PFEs in such a way that ensures that no PFE is overwhelmed by the anticipated peak load. As part of the network planning for your particular deployment, you must consider the exact mix of the distribution of the pseudowire logical interface devices and the services associated with the devices.

BEST PRACTICE: **A configured pseudowire logical interface device consumes resources from shared pools even when the device has no active subscriber logical interfaces. To conserve resources, do not deploy an excessive number of pseudowire devices that you do not intend to use.**

To configure the number of pseudowire logical interface devices that you want the router to support:

1. Specify that you want to configure the pseudowire service.

   ```
   [edit chassis]
   user@host# edit pseudowire-service
   ```

2. Set the maximum number of pseudowire logical interface devices.

   ```
   [edit chassis pseudowire-service]
   user@host# set device-count 500
   ```

Related Documentation

- Pseudowire Subscriber Logical Interfaces Overview on page 321
- Configuring a Pseudowire Subscriber Logical Interface on page 323

Configuring a Pseudowire Subscriber Logical Interface Device

To configure a pseudowire logical interface device that the router uses for subscriber logical interfaces, you specify the logical tunnel that processes the pseudowire termination. You can also configure additional optional parameters for the interface device, such as VLAN tagging method, MTU, and gratuitous ARP support.

To configure the pseudowire subscriber interface device:

1. Specify that you want to configure the pseudowire subscriber logical interface device.

   user@host# **edit interfaces** ps0

2. Specify the logical tunnel interface that is the anchor point for the pseudowire logical interface device. The anchor point must be an **lt** device in the format **lt-*fpc/pic/port***.

   NOTE: Tunnel services must be enabled on the **lt** interface that is the anchor point. You use the command, **set chassis fpc *slot-number* pic *pic-number* tunnel-services bandwidth *bandwidth*** to enable tunnel services.

   [edit interfaces ps0]
   user@host# **set** anchor-point **lt-1/0/10**

3. (Optional) Specify the MAC address for the pseudowire logical interface device.

   NOTE: You should ensure that you change the MAC address prior to passing traffic or binding subscribers on the pseudowire port. Changing the MAC address when the pseudowire port is active (for example, while an upper layer protocol is negotiating) can negatively impact network performance until adjacencies learn of the new MAC address.

   [edit interfaces ps0]
   user@host# **set** mac **00:11:22:33:44:55**

4. (Optional) Specify the VLAN tagging method used for the pseudowire logical interface device. You can specify single tagging, dual (stacked) tagging, mixed (flexible) tagging, or no tagging.

   [edit interfaces ps0]
   user@host# **set** flexible-vlan-tagging

   See *Enabling VLAN Tagging* for additional information about VLAN tagging.

5. (Optional) Specify the MTU for the pseudowire logical interface device. If you do not explicitly configure the MTU, the router uses the default value of 1500.

   [edit interfaces ps0]
   user@host# **set** mtu **2500**

   See *Setting the Protocol MTU* for additional information.

6. (Optional) Specify that the pseudowire logical interface device does not respond to gratuitous ARP requests.

   [edit interfaces ps0]
   user@host# **set** no-gratuitous-arp-request

   See *Configuring Gratuitous ARP* for additional information.

7. Configure additional optional parameters for the pseudowire logical interface device, such as *description*, *apply-groups*, *apply-groups-except*, and traceoptions.

Related Documentation

- Pseudowire Subscriber Logical Interfaces Overview on page 321
- Configuring a Pseudowire Subscriber Logical Interface on page 323
- *Configuring Tunnel Interfaces on MX Series Routers*
- *Router Chassis Configuration Statements*

Configuring the Transport Logical Interface for a Pseudowire Subscriber Logical Interface

This topic describes how to configure a pseudowire transport logical interface. A pseudowire device can have only one transport logical interface.

A pseudowire logical device and its related pseudowire logical interfaces are dependent on the state of the underlying logical transport interface device, which is either the Layer 2 VPN or Layer 2 circuit.

NOTE: We recommend that you use unit 0 to represent the transport logical interface for the pseudowire device. Non-zero unit numbers represent *service* logical interfaces used for pseudowire subscriber interfaces.

To configure a pseudowire transport logical interface:

1. Specify that you want to configure the pseudowire subscriber logical interface device.

   ```
   [edit]
   user@host# edit interfaces ps0
   ```

2. Specify that you want to configure unit 0, which represents the transport logical interface.

   ```
   [edit interfaces ps0]
   user@host# edit unit 0
   ```

3. Specify the **ethernet-ccc** encapsulation method for the transport logical interface.

   ```
   [edit interfaces ps0 unit 0]
   user@host# set encapsulation ethernet-ccc
   ```

Related Documentation

- Pseudowire Subscriber Logical Interfaces Overview on page 321
- Configuring a Pseudowire Subscriber Logical Interface on page 323

Configuring Layer 2 Circuit Signaling for Pseudowire Subscriber Logical Interfaces

This topic describes the steps for configuring Layer 2 circuit signaling used for the pseudowire subscriber logical interface support. You can also use Layer 2 VPN signaling for pseudowire subscriber logical interfaces. The two methods are mutually exclusive; you can use only one method for a particular pseudowire.

To configure Layer 2 circuit signaling for pseudowire interfaces:

1. Specify that you want to configure Layer 2 circuit parameters at the protocols hierarchy level.

   [edit protocols]
   user@host# **edit l2circuit**

2. Specify the IP address of the neighbor, to identify the PE router used for the Layer 2 circuit.

   [edit protocols l2circuit]
   user@host# **edit neighbor 192.168.102.15**

3. Specify the interface used by the Layer 2 circuit traffic.

   [edit protocols l2circuit neighbor 192.168.102.15]
   user@host# **edit interface ps1.0**

4. Configure the virtual circuit ID that identifies the Layer 2 circuit for the pseudowire.

   [edit protocols l2circuit neighbor 192.168.102.15 interface ps1.0]
   user@host# **set virtual-circuit-id 5**

5. (Optional) If multiple VLAN interfaces are carried over the pseudowire Layer 2 payload, configure the **no-vlan-id-validate** statement. This statement prevents VLAN validation in the signaling.

   [edit protocols l2circuit neighbor 192.168.102.15 interface ps1.0]
   user@host# **set no-vlan-id-validate**

For more information about Layer 2 circuits, see *Configuring Interfaces for Layer 2 Circuits*.

Related Documentation

- Pseudowire Subscriber Logical Interfaces Overview on page 321
- Configuring a Pseudowire Subscriber Logical Interface on page 323
- *Configuring Interfaces for Layer 2 Circuits*

Configuring Layer 2 VPN Signaling for Pseudowire Subscriber Logical Interfaces

This topic describes the steps for configuring Layer 2 VPN signaling used for the pseudowire subscriber logical interface support. You can also use Layer 2 circuit signaling for pseudowire subscriber logical interfaces. The two methods are mutually exclusive; you can use only one method on a particular pseudowire.

To configure Layer 2 VPN signaling for pseudowire interfaces:

1. Specify the name of the routing instance you want to configure.

```
[edit]
user@host# edit routing-instances l2vpn0
```

2. Configure the Layer 2 VPN routing instance type.

```
[edit routing-instances l2vpn0]
user@host# set instance-type l2vpn
```

3. Associate the pseudowire logical interface for the Layer 2 VPN.

```
[edit routing-instances l2vpn0]
user@host# set interface ps1.0
```

4. Configure the unique identifier for the routes that belong to the Layer 2 VPN.

```
[edit routing-instances l2vpn0]
user@host# set route-distinguisher 111.1.1.1:100
```

5. Configure the VPN routing and forwarding (VRF) target of the routing instance.

```
[edit routing-instances l2vpn0]
user@host# set vrf-target target:10:100
```

6. Specify that you want to configure the Layer 2 VPN protocol for the routing instance.

```
[edit routing-instances l2vpn0]
user@host# edit protocols l2vpn
```

7. Configure the encapsulation type for the routing instance.

```
[edit routing-instances l2vpn0 protocols l2vpn]
user@host# set encapsulation-type ethernet
```

8. Specify the site name and site identifier for the Layer 2 VPN.

```
[edit routing-instances l2vpn0 protocols l2vpn]
user@host# set site PE1 site-identifier 1
```

9. Specify the interface that connects to the site, and the remote interface to which you want the specified interface to connect.

```
[edit routing-instances l2vpn0 protocols l2vpn]
user@host# set interface ps1.0 remote-site-id 2
```

10. Configure the tracing options for traffic that uses the Layer 2 VPN.

```
[edit routing-instances l2vpn0 protocols l2vpn]
user@host# set traceoptions file l2vpn flag all
```

Related Documentation

- Pseudowire Subscriber Logical Interfaces Overview on page 321
- Configuring a Pseudowire Subscriber Logical Interface on page 323

Configuring the Service Logical Interface for a Pseudowire Subscriber Logical Interface

This topic describes how to configure a pseudowire service logical interface. Service logical interfaces represent the attachment circuits for pseudowire logical interfaces.

As described in the "Pseudowire Subscriber Logical Interfaces Overview" on page 321, you can choose whether to configure a service logical interface together with a higher subscriber logical interface, depending upon the business need. In a broadband edge configuration, the higher subscriber logical interface is the demarcation point for subscribers. However, in a business edge configuration, the service logical interface is the demarcation point for the business subscribers, and also serves as the subscriber logical interface, so no subscriber logical interfaces are explicitly configured.

NOTE: Non-zero unit numbers represent *service* logical interfaces used for pseudowire subscriber interfaces. Use unit 0 to represent the *transport* logical interface for the pseudowire device.

To configure a pseudowire service logical interface:

1. Specify that you want to configure the pseudowire subscriber logical interface device.

   ```
   [edit]
   user@host# edit interfaces ps0
   ```

2. Configure the unit for the service logical interface. Use a non-zero unit number.

   ```
   [edit interfaces ps0]
   user@host# edit unit 1
   ```

3. Configure the VLAN tag IDs.

   ```
   [edit interfaces ps0 unit 1]
   user@host# set vlan-tags outer 1 inner 1
   ```

4. Configure the interface to respond to ARP requests when the device has an active route to the ARP request target address.

   ```
   [edit interfaces ps0 unit 1]
   user@host# set proxy-arp
   ```

5. Specify that you want to configure the protocol family information. Pseudowire service logical interfaces support IPv4 (inet), IPv6 (inet6), and PPPoE (pppoe) protocol families.

   For example, to configure the IPv4 family:

   a. Specify that you want to configure IPv4.

      ```
      [edit interfaces ps0 unit 1]
      user@host# edit family inet
      ```

   b. Configure the parameters for the family.

      ```
      [edit interfaces ps0 unit 1 family inet]
      user@host# set filter input filter 1 output filter 4
      ```

```
user@host# set mac-validate loose
user@host# set input-hierarchical-policer policer-1
user@host# set unnumbered-address lo0.0 preferred-source-address 100.0.0.1
```

Related Documentation

- Pseudowire Subscriber Logical Interfaces Overview on page 321
- Configuring a Pseudowire Subscriber Logical Interface on page 323

CHAPTER 31

Configuring Hierarchical CoS Pseudowire Subscriber Interfaces

Hierarchical CoS on MPLS Pseudowire Subscriber Interfaces Overview

Junos OS supports two aspects of CoS for MPLS pseudowire subscriber interfaces. You can apply CoS rewrite rules and behavior aggregate (BA) classifiers to MPLS pseudowire subscriber interfaces. In addition, CoS performs egress hierarchical shaping towards the subscriber on MPLS pseudowire subscriber interfaces.

Hierarchical CoS enables you to apply traffic scheduling and queuing parameters and packet transmission scheduling parameters to an individual subscriber interface rather than to all interfaces configured on the port. Hierarchical CoS is supported on MX Series routers with either EQ DPCs or MPC/MICs installed.

On Juniper Networks MX Series routers, MPC/MIC and EQ DPC interfaces support a four-level CoS scheduling hierarchy that, when fully configured, consists of the physical interface (level 1), the interface set or the underlying interface (level 2), one or more logical interfaces (level 3), and one or more queues (level 4). Although all CoS scheduling hierarchies are four-level, level 1 is always the physical interface and level 4 is always the queue. Hierarchical scheduling configurations consist of the type of interfaces you configure; for example, a logical interface or an interface set and where those interfaces reside in the scheduling hierarchy, either level 2 or level 3. Because many hierarchical scheduling configurations are possible, we use the terms *two-level hierarchical scheduling* and *three-level hierarchical scheduling* in this discussion.

Related Documentation

- Pseudowire Subscriber Logical Interfaces Overview on page 321
- Configuring a Pseudowire Subscriber Logical Interface on page 323
- *Understanding Two-Level and Three-Level Hierarchical CoS for Subscriber Interfaces*
- CoS Two-Level Hierarchical Scheduling on MPLS Pseudowire Subscriber Interfaces on page 337
- CoS Three-Level Hierarchical Scheduling on MPLS Pseudowire Subscriber Interfaces on page 341

- CoS Configuration Overview for MPLS Pseudowire Subscriber Interfaces on page 334
- **hierarchical-scheduler (Subscriber Interfaces on MX Series Routers) on page 491**

CoS Configuration Overview for MPLS Pseudowire Subscriber Interfaces

CoS supports two-level and three-level hierarchies for MPLS pseudowire subscriber interfaces.

To configure two-level scheduling, include the **maximum-hierarchy-levels 2** option under the **[edit interfaces *interface-name* hierarchical-scheduler]** statement on the physical interface of the logical tunnel anchor point.

To configure three-level hierarchical scheduling, include the **implicit-hierarchy** option under the **[edit interfaces *interface-name* hierarchical-scheduler]** statement on the physical interface of the logical tunnel anchor point. Use the following guidelines for configuring the **implicit-hierarchy** option:

- If an output traffic-control profile is configured on the pseudowire transport interface and on a pseudowire service interface, the two interfaces form a scheduling hierarchy. The pseudowire transport interface resides in a level 2 scheduler node and the pseudowire service interface resides in a level 3 scheduler node.
- If an output traffic-control profile is configured on the pseudowire services interface but not on a pseudowire transport interface, the pseudowire services interface resides in a level 3 scheduler node.
- If an output traffic-control profile is only configured on the pseudowire transport interface and not on the pseudowire services interface, the pseudowire transport interface resides in a level 3 scheduler node and all pseudowire traffic uses this node.

If the **implicit-hierarchy** option is not set on the logical tunnel anchor point, logical interfaces behave normally with the hierarchical-scheduler mode configured with or without the **hierarchical-scheduler maximum-hierarchy-levels** option under the **[edit interfaces *interface-name* hierarchical-scheduler]** statement. In this case, when you apply a traffic-control profile to the pseudowire and service logical interfaces, they both reside in level 3 scheduler nodes and do not form a scheduling hierarchy, which might not be the desirable behavior. In business edge, where only the pseudowire logical interfaces need to be shaped, applying the traffic-control profile at just the transport logical interface may be sufficient.

When configuring the logical tunnel physical interface for the maximum hierarchy level, all pseudowire logical interfaces operating on the physical interface use the same hierarchy model. If you want to mix two-level and three-level scheduling hierarchies, you can group the pseudowires together by hierarchy levels and share the same logical tunnel anchor point or you can use three-level scheduling for all pseudowires over the anchor point.

To specify rewrite rules and classifiers on pseudowire interfaces, reference the pseudowire device under the **[edit class-of-service interfaces]** hierarchy level and specify the rewrite rules and classifiers for the pseudowire interfaces.

To control all pseudowire traffic using the same logical tunnel interface, apply CoS policies at the physical interface for the anchor logical tunnel.

Related Documentation

- Pseudowire Subscriber Logical Interfaces Overview on page 321
- Configuring a Pseudowire Subscriber Logical Interface on page 323
- *Understanding Two-Level and Three-Level Hierarchical CoS for Subscriber Interfaces*
- Hierarchical CoS on MPLS Pseudowire Subscriber Interfaces Overview on page 333
- Configuring CoS Two-Level Hierarchical Scheduling for MPLS Pseudowire Subscriber Interfaces on page 339
- Configuring CoS Three-Level Hierarchical Scheduling for MPLS Pseudowire Subscriber Interfaces (Logical Interfaces over a Transport Logical Interface) on page 344
- Configuring CoS Three-Level Hierarchical Scheduling for MPLS Pseudowire Subscriber Interfaces (Logical Interfaces over a Pseudowire Interface Set) on page 346
- **hierarchical-scheduler (Subscriber Interfaces on MX Series Routers) on page 491**

CHAPTER 32

Configuring CoS Two-Level Hierarchical Scheduling

CoS Two-Level Hierarchical Scheduling on MPLS Pseudowire Subscriber Interfaces

Two-level hierarchical scheduling limits the number of hierarchical levels in the scheduling hierarchy to two. In a two-level scheduling hierarchy, all logical interfaces and interface sets share a single level 2 node. Table 8 on page 337 summarizes the interface hierarchy and the CoS scheduler node levels for two-level hierarchical scheduling.

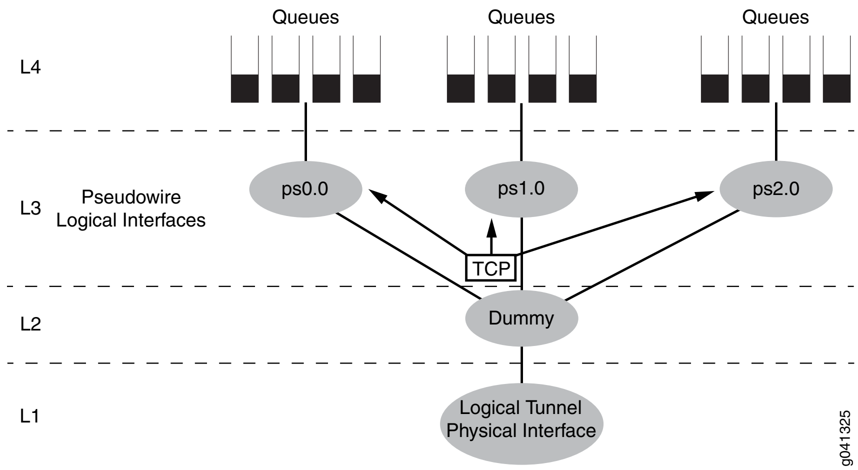
Table 8: Two-Level Hierarchical Scheduling–Interface Hierarchy Versus Scheduling Nodes

Level 1	Level 2	Level 3	Level 4
Physical interface	–	Pseudowire transport logical interface	One or more queues
Physical interface	–	Interface set	One or more queues
Physical interface	–	Pseudowire service logical interface	One or more queues

You use the two-level hierarchical scheduling when you have many pseudowires but you do not require shaping specific to the subscriber logical interface. For example, when your configuration is one subscriber per pseudowire interface.

Figure 23 on page 338 shows a two-level hierarchical scheduling configuration for the MPLS pseudowires. In this configuration, level 1 is the physical interface used for the logical tunnel anchor node. All of the pseudowire transport interfaces share a single level 2 node. The level 3 nodes are the pseudowire transport logical interfaces (ps0.0, ps1.0, and ps2.0). In this configuration, interface sets are not configured and only the logical interfaces have traffic control profiles.

Figure 23: MPLS Pseudowire Subscriber Interface Two-Level Scheduler Configuration



Two-level hierarchical scheduling has up to eight class of service queues. For this configuration, include the **maximum-hierarchy-levels 2** option under the **[edit interfaces *interface-name* hierarchical-scheduler]** statement at the physical interface for the anchor logical tunnel.

NOTE: You cannot configure shaping policies on both the pseudowire logical interfaces and the subscriber logical interfaces over the same pseudowire. If a traffic-control profile is configured on a pseudowire logical interface, and CoS policies are configured on the subscriber logical interface over another pseudowire, all of the logical interfaces are at level 3 and act as peers.

Related Documentation

- Pseudowire Subscriber Logical Interfaces Overview on page 321
- Configuring a Pseudowire Subscriber Logical Interface on page 323
- *Understanding Two-Level and Three-Level Hierarchical CoS for Subscriber Interfaces*
- Hierarchical CoS on MPLS Pseudowire Subscriber Interfaces Overview on page 333
- CoS Three-Level Hierarchical Scheduling on MPLS Pseudowire Subscriber Interfaces on page 341
- CoS Configuration Overview for MPLS Pseudowire Subscriber Interfaces on page 334
- Configuring CoS Two-Level Hierarchical Scheduling for MPLS Pseudowire Subscriber Interfaces on page 339
- **hierarchical-scheduler (Subscriber Interfaces on MX Series Routers) on page 491**

Configuring CoS Two-Level Hierarchical Scheduling for MPLS Pseudowire Subscriber Interfaces

Before configuring CoS parameters for MPLS pseudowire subscriber interfaces, you must first complete these tasks:

1. Configure the pseudowire logical interfaces. See "Configuring a Pseudowire Subscriber Logical Interface" on page 323.
2. Configure the pseudowire device count. See "Configuring the Maximum Number of Pseudowire Logical Interface Devices Supported on the Router" on page 325.
3. Configure the pseudowire device including the logical tunnel anchor point. See "Configuring a Pseudowire Subscriber Logical Interface Device" on page 325.
4. Configure the pseudowire transport logical interface. See "Configuring the Transport Logical Interface for a Pseudowire Subscriber Logical Interface" on page 327.
5. Configure the pseudowire signaling (either Layer 2 circuit signaling or Layer 2 VPN signaling). See "Configuring Layer 2 Circuit Signaling for Pseudowire Subscriber Logical Interfaces" on page 328 or "Configuring Layer 2 VPN Signaling for Pseudowire Subscriber Logical Interfaces" on page 328.
6. Configure the pseudowire logical interfaces. See "Configuring the Service Logical Interface for a Pseudowire Subscriber Logical Interface" on page 330.

To configure CoS policies on MPLS pseudowire subscriber interfaces using two-level scheduling:

1. Configure the hierarchical scheduler for the physical interface used for the logical tunnel (anchor point). For two-level scheduling the hierarchical scheduler must be set to **maximum-scheduler levels 2**.

   [edit]
   user@host# **edit interfaces ps** ***ps-anchor-device-name***
   user@host# **set hierarchical-scheduler maximum-hierarchy-levels 2**

2. Specify the traffic-control profile to use on the pseudowire logical interface.

   [edit class-of-service]
   user@host# **edit interfaces ps** ***ps-device-name***
   user@host# **edit unit** ***logical-unit-number***
   user@host# **set output-traffic-control-profile** ***profile-name***

3. Configure the rewrite rule.

   The available rewrite rule types for pseudowire interfaces are **dscp** and **inet-precedence**.

   [edit class-of-service]
   user@host# **edit interfaces ps** ***ps-device-name***
   user@host# **edit unit** ***logical-unit-number***
   user@host# **edit rewrite-rules (dscp | inet-precedence)** ***rewrite-name***
   user@host# **edit forwarding-class** ***class-name***
   user@host# **set loss-priority** ***class-name*** **code-point (*alias* | *bits*)**

4. Configure the classifier.

The available classifier types for pseudowire interfaces are **dscp** and **inet-precedence**.

[edit class-of-service]
user@host# **edit interfaces ps** ***ps-device-name***
user@host# **edit unit** ***logical-unit-number***
user@host# **edit classifiers (dscp | inet-precedence)** ***classifier-name***
user@host# **edit forwarding-class** ***class-name***
user@host# **set loss-priority** ***class-name*** **code-points [*aliases*] [*bit-patterns*]**

5. Apply the rewrite rule and classifier to the pseudowire interface.

   For the *interface_name* parameter, specify the pseudowire device name.

   [edit class-of-service interfaces *interface_name* unit *logical-unit-number*]
   user@host# **set rewrite-rule (dscp | inet-precedence) (*rewrite-name* | default) protocol *protocol-types***
   user@host# **set classifiers (dscp | inet-precedence) (*classifier-name* | default)**

Related Documentation

- *CoS on Ethernet Pseudowires in Universal Edge Networks Overview*
- *CoS Inputs and Outputs Examples*
- *Understanding Two-Level and Three-Level Hierarchical CoS for Subscriber Interfaces*
- Hierarchical CoS on MPLS Pseudowire Subscriber Interfaces Overview on page 333
- CoS Two-Level Hierarchical Scheduling on MPLS Pseudowire Subscriber Interfaces on page 337
- CoS Three-Level Hierarchical Scheduling on MPLS Pseudowire Subscriber Interfaces on page 341
- CoS Configuration Overview for MPLS Pseudowire Subscriber Interfaces on page 334
- **hierarchical-scheduler (Subscriber Interfaces on MX Series Routers) on page 491**

CHAPTER 33

Configuring CoS Three-Level Hierarchical Scheduling

CoS Three-Level Hierarchical Scheduling on MPLS Pseudowire Subscriber Interfaces

In three-level hierarchical scheduling, the CoS scheduler nodes at level 1, level 2, and level 3 form a scheduling hierarchy. You can configure many different three-level scheduling hierarchies, depending on the location of the interface set and the use of underlying interfaces. In all variations, the physical interface on which the logical tunnel resides is a level 1 CoS scheduler node and the queues reside at level 4. Three-level scheduling hierarchies can have up to eight class of service queues.
Table 9 on page 341summarizes the most common three-level hierarchical scheduling configurations and shows the interface hierarchy and CoS scheduler nodes.

Table 9: Three-Level Hierarchical Scheduling–Interface Hierarchy Versus CoS Scheduling Node Levels

Level 1	Level 2	Level 3	Level 4
Physical interface	Pseudowire interface set	Pseudowire service logical interfaces	One or more queues
Physical interface	Pseudowire transport logical interface	Pseudowire interface set	One or more queues
Physical interface	Pseudowire transport logical interface	Pseudowire service logical interfaces	One or more queues

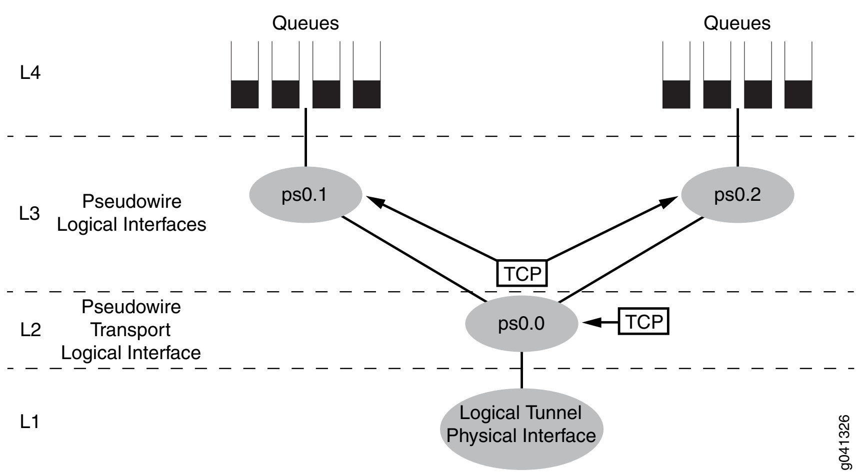
Three-Level Scheduling Hierarchy: Pseudowire Logical Interfaces over a Transport Logical Interface

Figure 24 on page 342 shows an MPLS pseudowire three-level scheduling hierarchy that includes two pseudowire service logical interfaces over a pseudowire transport logical interface. This variation uses the following scheduler nodes:

- Level 4—Forwarding class-based queues
- Level 3—Pseudowire service logical interfaces (ps0.1 and ps0.2) for subscriber sessions
- Level 2—Pseudowire transport logical interface (ps0.0)
- Level 1—Common/shared physical interface of the logical tunnel anchor point

You apply the traffic-control profiles at the pseudowire transport logical interfaces (level 2) and the pseudowire service logical interfaces (level 3).

Figure 24: Three-Level Scheduling Hierarchy Case 1: Pseudowire Service Logical Interfaces over a Transport Logical Interface



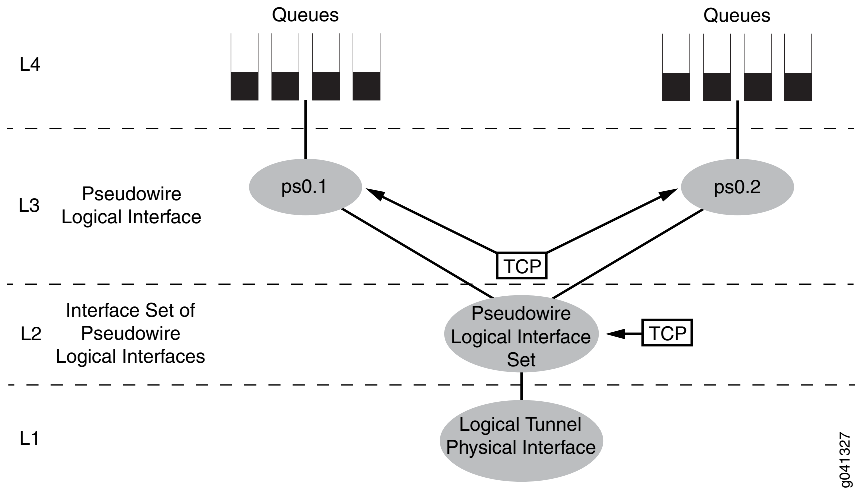
Three-Level Scheduling Hierarchy : Pseudowire Service Logical Interfaces over a Pseudowire Service Interface Set

Figure 25 on page 343 shows another variation of MPLS pseudowire three-level hierarchical scheduling that includes two pseudowire service logical interfaces over a pseudowire service interface set. This variation uses the following CoS scheduler nodes:

- Level 4—Forwarding class-based queues
- Level 3—Pseudowire service logical interfaces (ps0.1 and ps0.2)
- Level 2—Pseudowire service interface set
- Level 1—Common/shared physical interface of the logical tunnel anchor point

You apply the traffic-control profile at the pseudowire service interfaces (level 3) and at the interface set (level 2). This variation is most useful for subscriber edge deployments.

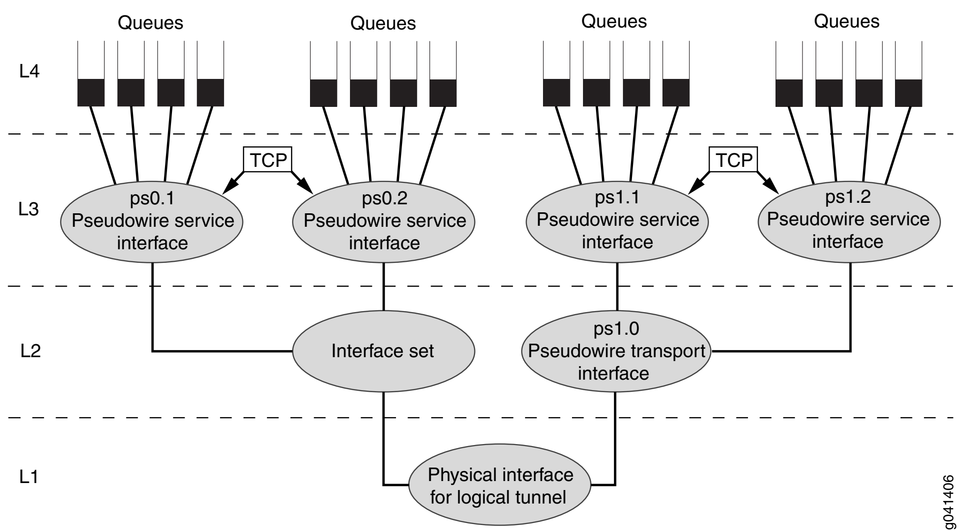
Figure 25: Three-Level Scheduling Hierarchy Case 2: Pseudowire Service Logical Interfaces over a Pseudowire Service Interface Set



Three-Level Scheduling Hierarchy Combined Deployment Scenario

Figure 26 on page 343 shows a deployment scenario that combines the three-level hierarchical scheduling scenarios in Figure 24 on page 342 and Figure 25 on page 343.

Figure 26: Three-Level Hierarchical Scheduling for MPLS Pseudowire Subscriber Interfaces—Deployment Scenario



This variation uses the following CoS scheduler nodes:

- Level 4—Forwarding class-based queues
- Level 3—Pseudowire service logical interfaces (ps0.1, ps0.2, ps1.1, and ps1.2)

- Level 2—Service interface set for pseudowire service interfaces (ps0.1 and ps0.2) and transport logical interface (ps1.0) for the pseudowire service logical interfaces (ps1.1 and ps1.2)
- Level 1—Common/shared physical interface of the logical tunnel anchor point

You apply the traffic-control profiles to the interfaces at both level 2 and level 3, as well as the interface set at level 2.

Related Documentation

- Pseudowire Subscriber Logical Interfaces Overview on page 321
- Configuring a Pseudowire Subscriber Logical Interface on page 323
- *Understanding Two-Level and Three-Level Hierarchical CoS for Subscriber Interfaces*
- Hierarchical CoS on MPLS Pseudowire Subscriber Interfaces Overview on page 333
- CoS Configuration Overview for MPLS Pseudowire Subscriber Interfaces on page 334
- Configuring CoS Three-Level Hierarchical Scheduling for MPLS Pseudowire Subscriber Interfaces (Logical Interfaces over a Transport Logical Interface) on page 344
- Configuring CoS Three-Level Hierarchical Scheduling for MPLS Pseudowire Subscriber Interfaces (Logical Interfaces over a Pseudowire Interface Set) on page 346
- **hierarchical-scheduler (Subscriber Interfaces on MX Series Routers) on page 491**

Configuring CoS Three-Level Hierarchical Scheduling for MPLS Pseudowire Subscriber Interfaces (Logical Interfaces over a Transport Logical Interface)

Before configuring CoS three-level scheduling on pseudowire logical interfaces over a transport logical interface, you must first complete these tasks:

1. Configure the pseudowire logical interfaces. See "Configuring a Pseudowire Subscriber Logical Interface" on page 323.
2. Configure the pseudowire device count. See "Configuring the Maximum Number of Pseudowire Logical Interface Devices Supported on the Router" on page 325.
3. Configure the pseudowire device including the logical tunnel anchor point. See "Configuring a Pseudowire Subscriber Logical Interface Device" on page 325.
4. Configure the pseudowire transport logical interface. See "Configuring the Transport Logical Interface for a Pseudowire Subscriber Logical Interface" on page 327.
5. Configure the pseudowire signaling (either Layer 2 circuit signaling or Layer 2 VPN signaling). See "Configuring Layer 2 Circuit Signaling for Pseudowire Subscriber Logical Interfaces" on page 328 or "Configuring Layer 2 VPN Signaling for Pseudowire Subscriber Logical Interfaces" on page 328.
6. Configure the pseudowire logical interfaces. See "Configuring the Service Logical Interface for a Pseudowire Subscriber Logical Interface" on page 330.

Three-level scheduling on pseudowire logical interfaces over a transport logical interface requires you to apply the traffic-control profiles at both the pseudowire logical interface and the pseudowire transport logical interface. To configure CoS policies on three-level scheduling on pseudowire logical interfaces over a transport logical interface:

1. Configure the hierarchical scheduler for the physical interface used for the logical tunnel (anchor point). For three-level scheduling the hierarchical scheduler must be set to **implicit-hierarchy**.

   [edit]
   user@host#**edit interfaces** ***ps-anchor-device-name***
   user@host#**set hierarchical-scheduler implicit-hierarchy**

2. Specify the traffic-control profile to use on the pseudowire logical interface.

   [edit class-of-service]
   user@host#**edit interfaces ps** ***ps-device-name***
   user@host#**edit unit** ***logical-unit-number***
   user@host#**set output-traffic-control-profile** ***profile-name***

3. Specify the traffic-control profile to use on the pseudowire transport logical interface.

   [edit class-of-service]
   user@host#**edit interfaces ps** ***ps-device-name***
   user@host#**edit unit** ***logical-unit-number***
   user@host#**set output-traffic-control-profile** ***profile-name***

4. Configure the rewrite rule.

   The available rewrite rule types for pseudowire interfaces are **dscp** and **inet-precedence**.

   [edit class-of-service]
   user@host#**edit interfaces ps** ***ps-device-name***
   user@host#**edit unit** ***logical-unit-number***
   user@host#**edit rewrite-rules (dscp | inet-precedence)** ***rewrite-name***
   user@host#**edit forwarding-class** ***class-name***
   user@host#**set loss-priority** ***class-name*** **code-point (*alias* | *bits*)**

5. Configure the classifier.

   The available classifier types for pseudowire interfaces are **dscp** and **inet-precedence**.

   [edit class-of-service]
   user@host#**edit interfaces ps** ***ps-device-name***
   user@host#**edit unit** ***logical-unit-number***
   user@host#**edit classifiers (dscp | inet-precedence)** ***classifier-name***
   user@host#**edit forwarding-class** ***class-name***
   user@host#**set loss-priority** ***class-name*** **code-points [*aliases*] [*bit-patterns*]**

6. Apply the rewrite rule and classifier to the pseudowire interfaces.

   For the *interface_name* parameter, specify the pseudowire device name.

   [edit class-of-service interfaces *interface_name* unit *logical-unit-number*]
   user@host#**set rewrite-rule (dscp | inet-precedence) (*rewrite-name* | default) protocol *protocol-types***
   user@host#**set classifiers (dscp | inet-precedence) (*classifier-name* | default)**

Related Documentation

- *CoS on Ethernet Pseudowires in Universal Edge Networks Overview*
- *CoS Inputs and Outputs Examples*
- *Understanding Two-Level and Three-Level Hierarchical CoS for Subscriber Interfaces*
- Hierarchical CoS on MPLS Pseudowire Subscriber Interfaces Overview on page 333
- CoS Three-Level Hierarchical Scheduling on MPLS Pseudowire Subscriber Interfaces on page 341
- CoS Configuration Overview for MPLS Pseudowire Subscriber Interfaces on page 334
- Configuring CoS Three-Level Hierarchical Scheduling for MPLS Pseudowire Subscriber Interfaces (Logical Interfaces over a Pseudowire Interface Set) on page 346
- **hierarchical-scheduler (Subscriber Interfaces on MX Series Routers) on page 491**

Configuring CoS Three-Level Hierarchical Scheduling for MPLS Pseudowire Subscriber Interfaces (Logical Interfaces over a Pseudowire Interface Set)

Before configuring three-level scheduling on pseudowire logical interfaces over a pseudowire logical interface set, you must first complete the following tasks:

1. Configure the pseudowire logical interfaces. See "Configuring a Pseudowire Subscriber Logical Interface" on page 323.
2. Configure the pseudowire device count. See "Configuring the Maximum Number of Pseudowire Logical Interface Devices Supported on the Router" on page 325.
3. Configure the pseudowire device including the logical tunnel anchor point. See "Configuring a Pseudowire Subscriber Logical Interface Device" on page 325.
4. Configure the pseudowire transport logical interface. See "Configuring the Transport Logical Interface for a Pseudowire Subscriber Logical Interface" on page 327.
5. Configure the pseudowire signaling (either Layer 2 circuit signaling or Layer 2 VPN signaling). See "Configuring Layer 2 Circuit Signaling for Pseudowire Subscriber Logical Interfaces" on page 328 or "Configuring Layer 2 VPN Signaling for Pseudowire Subscriber Logical Interfaces" on page 328.
6. Configure the pseudowire logical interfaces. See "Configuring the Service Logical Interface for a Pseudowire Subscriber Logical Interface" on page 330.

Three-level scheduling on pseudowire logical interfaces over a pseudowire logical interface set requires you to apply the traffic-control profiles at both the pseudowire logical interface and the pseudowire logical interface-set. To configure CoS policies on MPLS pseudowire subscriber interfaces using three-level implicit hierarchical scheduling:

1. Configure the hierarchical scheduler for the physical interface used for the logical tunnel (anchor point). For three-level scheduling the hierarchical scheduler must be set to **implicit-hierarchy**.

   [edit]
   user@host# **edit interfaces** ***ps-anchor-device-name***
   user@host# **set hierarchical-scheduler implicit-hierarchy**

2. Specify the traffic-control profile to use on the pseudowire logical interfaces.

   [edit class-of-service]
   user@host#**edit interfaces ps** ***ps-device-name***
   user@host#**edit unit** ***logical-unit-number***
   user@host#**set output-traffic-control-profile** ***profile-name***

3. Define a pseudowire logical interface set and configure the traffic-control profile used for the interface set.

   [edit class-of-service]
   user@host#**edit interfaces**
   user@host#**edit interface-set** ***interface-set-name***
   user@host#**edit output-traffic-control-profile** ***profile-name***

4. Group the pseudowire logical interfaces in the pseudowire logical interface set.

   [edit]
   user@host#**edit interfaces**
   user@host#**edit interface-set** ***interface-set-name***
   user@host#**edit interface ps** ***ps-device-name***
   user@host#**edit unit** ***logical-unit-number***

5. Configure the rewrite rule.

   The available rewrite rule types for pseudowire interfaces are **dscp** and **inet-precedence**.

   [edit class-of-service]
   user@host#**edit interfaces ps** ***ps-device-name***
   user@host#**edit unit** ***logical-unit-number***
   user@host#**edit rewrite-rules (dscp | inet-precedence)** ***rewrite-name***
   user@host#**edit forwarding-class** ***class-name***
   user@host#**set loss-priority** ***class-name*** **code-point (*alias* | *bits*)**

6. Configure the classifier.

   The available classifier types for pseudowire interfaces are **dscp** and **inet-precedence**.

   [edit class-of-service]
   user@host#**edit interfaces ps** ***ps-device-name***
   user@host#**edit unit** ***logical-unit-number***
   user@host#**edit classifiers (dscp | inet-precedence)** ***classifier-name***
   user@host#**edit forwarding-class** ***class-name***
   user@host#**set loss-priority** ***class-name*** **code-points [*aliases*] [*bit-patterns*]**

7. Apply the rewrite rule and classifier to the pseudowire interfaces.

   For the *interface_name* parameter, specify the ps device name.

   [edit class-of-service interfaces *interface_name* unit *logical-unit-number*]
   user@host#**set rewrite-rule (dscp | inet-precedence) (*rewrite-name* | default) protocol *protocol-types***
   user@host#**set classifiers (dscp | inet-precedence) (*classifier-name* | default)**

Related Documentation

- *CoS on Ethernet Pseudowires in Universal Edge Networks Overview*
- *CoS Inputs and Outputs Examples*
- *Understanding Two-Level and Three-Level Hierarchical CoS for Subscriber Interfaces*
- Hierarchical CoS on MPLS Pseudowire Subscriber Interfaces Overview on page 333

- CoS Three-Level Hierarchical Scheduling on MPLS Pseudowire Subscriber Interfaces on page 341
- CoS Configuration Overview for MPLS Pseudowire Subscriber Interfaces on page 334
- Configuring CoS Three-Level Hierarchical Scheduling for MPLS Pseudowire Subscriber Interfaces (Logical Interfaces over a Transport Logical Interface) on page 344
- **hierarchical-scheduler (Subscriber Interfaces on MX Series Routers) on page 491**

PART 6

Configuring ATM for Subscriber Access

CHAPTER 34

Configuring ATM to Deliver Subscriber-Based Services

ATM for Subscriber Access Overview

By using the ATM Modular Interface Card (MIC) with small form-factor pluggable transceiver (SFP) and a supported Modular Port Concentrator (MPC), you can configure the MX Series router to support configurations that enable subscribers to access the router over an ATM network using ATM Adaptation Layer 5 (AAL5) permanent virtual connections (PVCs). Using these configurations enables the delivery of subscriber-based services, such as class of service (CoS) and firewall filters, for subscribers accessing the router over an ATM network.

This overview describes the following topics about configuring ATM interfaces for subscriber access:

Supported Configurations for ATM Subscriber Access

On MX Series routers with MPC/MIC interfaces that use the ATM MIC with SFP (Model Number MIC-3D-8OC3-2OC12-ATM), you can create the following configurations to enable subscribers to access the router over an ATM network using ATM Adaptation Layer 5 (AAL5) permanent virtual connections (PVCs):

- PPP-over-Ethernet-over-ATM
- Routed IP-over-ATM
- Bridged IP-over-Ethernet-over-ATM
- PPP-over-ATM

The following sections briefly describe each supported ATM subscriber access configuration.

PPP-over-Ethernet-over-ATM Configurations

PPP-over-Ethernet-over-ATM (PPPoE-over-ATM) configurations support both statically created and dynamically created PPPoE (**pp0**) logical subscriber interfaces over static ATM underlying interfaces. Most PPPoE and subscriber services features supported on terminated connections and tunneled (L2TP access concentrator, or LAC) connections are also supported for access to an MX Series router over an ATM network.

PPPoE-over-ATM configurations require static configuration of the underlying ATM physical interface and ATM logical interface. You can configure the PPPoE (**pp0**) subscriber interface either dynamically, by means of a dynamic profile, or statically at the **[edit interfaces pp0 unit *logical-unit-number*]** hierarchy level.

For PPPoE-over-ATM configurations on an MX Series router, you must configure the ATM underlying interface with PPPoE-over-ATM logical link control (LLC) encapsulation. To do so, include the **encapsulation ppp-over-ether-over-atm-llc** statement at the **[edit interfaces *interface-name* unit *logical-unit-number*]** hierarchy level.

Using dynamic PPPoE-over-ATM configurations for ATM subscriber access enables you to configure an MX Series router to dynamically create PPPoE logical subscriber interfaces over static ATM underlying interfaces only when needed; that is, when a subscriber logs in on the associated underlying interface. Dynamic PPPoE over static ATM configurations are *not* supported on M Series routers and T Series routers.

Optionally, you can dynamically or statically apply subscriber services such as class of service (CoS) and firewall filters to the PPPoE (**pp0**) subscriber interface. For PPPoE-over-ATM configurations that create a dynamic PPPoE subscriber interface, you can configure CoS attributes and firewall filters in the dynamic profile that defines the **pp0** subscriber interface. For PPPoE-over-ATM configurations that create a static PPPoE subscriber interface, you can statically configure CoS attributes and firewall filters as you would for any static interface configured on an MX Series router.

Routed IP-over-ATM Configurations

Routed IP-over-ATM (IPoA) configurations support statically created IPv4 and IPv6 logical subscriber interfaces over static ATM underlying interfaces. IPoA configurations are typically used to implement business digital subscriber line (DSL) connections that do not require connection negotiation for address assignment.

IPoA configurations on MX Series routers require static configuration of the ATM underlying interface, IPv4 interface, IPv6 interface, CoS attributes, and firewall fiters. Dynamic configuration of these components is not supported.

To configure IPoA subscriber access on MX Series routers, you must configure either of the following encapsulation types on the ATM underlying interface:

- For IPoA encapsulation with logical link control (LLC), configure ATM subnetwork attachment point (SNAP) encapsulation by including the **encapsulation atm-snap** statement at the **[edit interfaces *interface-name* unit *logical-unit-number*]** hierarchy level.
- For IPoA encapsulation with virtual circuit (VC) multiplexing, configure ATM VC multiplex encapsulation by including the **encapsulation atm-vc-mux** statement at the **[edit interfaces *interface-name* unit *logical-unit-number*]** hierarchy level.

Optionally, you can statically configure subscriber services such as CoS and firewall filters and apply them to the IPv4 or IPv6 interface; you *cannot* use a dynamic profile for this purpose.

Bridged IP-over-Ethernet-over-ATM Configurations

Bridged IP-over-Ethernet-over-ATM (IPoE-over-ATM) configurations support statically created IPv4 and IPv6 logical subscriber interfaces over static ATM underlying interfaces. Like IPoA configurations, IPoE-over-ATM configurations are typically used in topologies that do not require connection negotiation for address assignment.

IPoE-over-ATM configurations on MX Series routers require static configuration of the ATM underlying interface, IP interface, CoS attributes, and firewall fiters. Dynamic configuration of these components is not supported.

For bridged IP-over-Ethernet-over-ATM configurations on an MX Series router, you must configure the ATM underlying interface with Ethernet-over-ATM LLC encapsulation. To do so, include the **encapsulation ether-over-atm-llc** statement at the **[edit interfaces *interface-name* unit *logical-unit-number*]** hierarchy level.

Optionally, you can statically configure subscriber services such as class of service (CoS) and firewall filters and apply them to the IPv4 or IPv6 interface; you *cannot* use a dynamic profile for this purpose.

PPP-over-ATM Configurations

PPP-over-ATM (PPPoA) configurations support statically created PPP logical subscriber interfaces over static ATM underlying interfaces. Most features supported for PPPoE configurations are also supported for PPP access to an MX Series router over an ATM network.

PPPoA configurations on MX Series routers require static configuration of the ATM underlying interface and PPP subscriber interface.

To configure PPPoA subscriber access on MX Series routers, you must configure either of the following encapsulation types on each PPP logical subscriber interface:

- For PPPoA encapsulation with logical link control (LLC), configure PPP-over-AAL5 LLC encapsulation by including the **encapsulation atm-ppp-llc** statement at the **[edit interfaces *interface-name* unit *logical-unit-number*]** hierarchy level.
- For PPPoA encapsulation with virtual circuit (VC) multiplexing, configure PPP-over-AAL5 multiplex encapsulation by including the **encapsulation atm-ppp-vc-mux** statement at the **[edit interfaces *interface-name* unit *logical-unit-number*]** hierarchy level.

Optionally, you can use dynamic profiles to dynamically or statically apply subscriber services, such as CoS attributes and firewall filters, to the static PPP subscriber interface. Configuring CoS and firewall filters in this manner enables you to efficiently and economically provide these services to PPP subscribers accessing the router over an ATM network.

Configuration and Encapsulation Types for ATM Subscriber Access

You use the same basic statements, commands, and procedures to create, verify, and manage PPPoE-over-ATM, IPoA, IPoE-over-ATM, and PPPoA configurations as the statements, commands, and procedures you use for static configurations on M Series routers and T Series routers, and for dynamic PPPoE configurations on MX Series routers.

A critical element of configuring ATM subscriber access is ensuring that you specify the correct encapsulation type for the ATM logical interface. The encapsulation type you use depends on the supported configuration and, for IPoA and PPPoA configurations, whether you want to configure an encapsulation type that uses logical link control (LLC) or virtual circuit (VC) multiplexing.

Related Documentation

- ATM for Subscriber Access Encapsulation Types Overview on page 355
- Configuring ATM for Subscriber Access on page 358
- Example: Configuring a Dynamic PPPoE Subscriber Interface over ATM on page 369
- Example: Configuring a Static PPPoE Subscriber Interface over ATM on page 378
- Example: Configuring a Static Subscriber Interface for IP Access over ATM on page 387
- Example: Configuring a Static Subscriber Interface for IP Access over Ethernet over ATM on page 393
- Example: Configuring a Static PPP Subscriber Interface over ATM on page 399

ATM for Subscriber Access Encapsulation Types Overview

To enable subscriber access to an MX Series router over an ATM network, you can create any of the following configurations on Modular Port Concentrator/Modular Interface Card (MPC/MIC) interfaces that use the ATM MIC with SFP:

- PPP-over-Ethernet-over-ATM (PPPoE-over ATM) with a dynamic or static PPPoE (**pp0**) subscriber interface over a static ATM underlying interface
- Routed IP-over-ATM (IPoA) with a static IPv4 or IPv6 subscriber interface over a static ATM underlying interface
- Bridged IP-over-Ethernet-over-ATM (IPoE-over-ATM) with a static IPv4 or IPv6 subscriber interface over a static ATM underlying interface
- PPP-over-ATM (PPPoA) with a static PPP subscriber interface over a static ATM underlying interface

As part of the configuration procedure, you must specify the appropriate encapsulation type for your configuration on the ATM logical interface.

Table 10 on page 355 lists and describes the encapsulation type you must specify as part of the **encapsulation** statement when you configure the ATM logical interface for each supported configuration.

Table 10: Encapsulation Types for Supported ATM Subscriber Access Configurations

ATM Subscriber Access Configuration	Encapsulation Type	Description
PPPoE-over-ATM with dynamic **pp0** subscriber interface	**ppp-over-ether-over-atm-llc**	PPPoE-over-ATM encapsulation with logical link control (LLC)
PPPoE-over-ATM with static **pp0** subscriber interface	**ppp-over-ether-over-atm-llc**	PPPoE-over-ATM encapsulation with LLC
IP-over-ATM (IPoA)	**atm-snap**	ATM subnetwork attachment point (SNAP) encapsulation for IPoA with LLC
	atm-vc-mux	ATM VC multiplex encapsulation for IPoA with virtual circuit (VC) multiplexing
IP-over-Ethernet-over-ATM (IPoE-over-ATM)	**ether-over-atm-llc**	Ethernet-over-ATM encapsulation with LLC

Table 10: Encapsulation Types for Supported ATM Subscriber Access Configurations *(continued)*

ATM Subscriber Access Configuration	Encapsulation Type	Description
PPP-over-ATM (PPPoA)	**atm-ppp-llc** (for PPPoA with logical link control)	PPP-over-AAL5 encapsulation with LLC
	atm-ppp-vc-mux (for PPPoA with virtual circuit multiplexing)	PPP-over-AAL5 encapsulation with VC multiplexing

Related Documentation

- ATM for Subscriber Access Overview on page 351
- Configuring ATM for Subscriber Access on page 358
- Example: Configuring a Dynamic PPPoE Subscriber Interface over ATM on page 369
- Example: Configuring a Static PPPoE Subscriber Interface over ATM on page 378
- Example: Configuring a Static Subscriber Interface for IP Access over ATM on page 387
- Example: Configuring a Static Subscriber Interface for IP Access over Ethernet over ATM on page 393
- Example: Configuring a Static PPP Subscriber Interface over ATM on page 399

Guidelines for Configuring ATM for Subscriber Access

The following guidelines apply when you configure PPP-over-Ethernet-over-ATM (PPPoE-over-ATM), IP-over-ATM (IPoA), IP-over-Ethernet-over-ATM (IPoE-over-ATM), or PPP-over-ATM (PPPoA) configurations for ATM subscriber access. You can create these configurations on MX Series routers with Modular Port Concentrator/Modular Interface Card (MPC/MIC) interfaces that use the ATM MIC with SFP.

For all supported ATM subscriber access configurations:

- Make sure you specify the correct encapsulation type on the ATM logical interface for your configuration, as described in “ATM for Subscriber Access Encapsulation Types Overview” on page 355.

For PPPoE-over-ATM configurations:

- For dynamic or static PPPoE-over-ATM configurations, specify PPPoE-specific options at the **[edit interfaces *interface-name* unit *logical-unit-number* family pppoe]** hierarchy level. Specifying PPPoE-specific options at the **[edit interfaces *interface-name* unit *logical-unit-number* pppoe-underlying-options]** hierarchy level is not supported for these configurations.
- For dynamic or static PPPoE-over-ATM configurations, you must configure the router to act as a PPPoE server (also known as a *remote access concentrator*). Configuring the router to act as a PPPoE client is not supported in these configurations.

- For dynamic PPPoE-over-ATM configurations, issue the **dynamic-profile *profile-name*** statement at the **[edit interfaces *interface-name* unit *logical-unit-number* family pppoe]** hierarchy level to associate the ATM logical interface with the dynamic profile that defines the PPPoE subscriber interface.

For static IPoA and IPoE-over-ATM configurations:

- Specify interface-specific options at the **[edit interfaces *interface-name* unit *logical-unit-number* family inet]** hierarchy level (for IPv4) or at the **[edit interfaces *interface-name* unit *logical-unit-number* family inet6]** hierarchy level (for IPv6).

For static PPPoA configurations:

- Specify PPP-specific options at the **[edit interfaces *interface-name* unit *logical-unit-number* ppp-options]** hierarchy level.

Related Documentation

- ATM for Subscriber Access Overview on page 351
- ATM for Subscriber Access Encapsulation Types Overview on page 355
- Configuring ATM for Subscriber Access on page 358
- Example: Configuring a Dynamic PPPoE Subscriber Interface over ATM on page 369
- Example: Configuring a Static PPPoE Subscriber Interface over ATM on page 378
- Example: Configuring a Static Subscriber Interface for IP Access over ATM on page 387
- Example: Configuring a Static Subscriber Interface for IP Access over Ethernet over ATM on page 393
- Example: Configuring a Static PPP Subscriber Interface over ATM on page 399

Configuring ATM for Subscriber Access

On MX Series routers with MPC/MIC interfaces that use the ATM MIC with SFP, you can create the following configurations to enable subscribers to access the router over an ATM network using ATM Adaptation Layer 5 (AAL5) permanent virtual connections (PVCs):

- PPP-over-Ethernet-over-ATM (PPPoE-over ATM) with a dynamic PPPoE (**pp0**) subscriber interface over a static ATM underlying interface
- PPP-over-Ethernet-over-ATM (PPPoE-over ATM) with a static PPPoE (**pp0**) subscriber interface over a static ATM underlying interface
- Routed IP-over-ATM (IPoA) with a static IPv4 or IPv6 subscriber interface over a static ATM underlying interface
- Bridged IP-over-Ethernet-over-ATM with a static IPv4 or IPv6 subscriber interface over a static ATM underlying interface
- PPP-over-ATM (PPPoA) with a static PPP subscriber interface over a static ATM underlying interface

Before you begin:

1. Make sure the MX Series router you are using has Modular Port Concentrator/Modular Interface Card (MPC/MIC) interfaces and an ATM MIC with SFP (Model Number MIC-3D-8OC3-2OC12-ATM) installed and operational.
   - For information about compatible MPCs for the ATM MIC with SFP, see the *MX Series Interface Module Reference*.
   - For information about installing MPCs and MICs in an MX Series router, see the *Hardware Guide* for your MX Series router model.
2. Make sure you understand how to configure and use static ATM interfaces.

   See *ATM Interfaces Overview*.
3. If your configuration includes dynamic profiles for PPPoE, class of service (CoS) attributes, or standard firewall filters, make sure you understand how to configure these attributes and apply them to the subscriber interface.
   - For PPPoE dynamic profiles, see "Configuring Dynamic PPPoE Subscriber Interfaces Using Dynamic Profiles" on page 145
   - For CoS configuration, see *Configuring Traffic Scheduling and Shaping for Subscriber Access*
   - For standard firewall flter configuration, see *Guidelines for Configuring Firewall Filters* and *Guidelines for Applying Firewall Filters*

To configure ATM for subscriber access on an MX Series router:

1. For a PPPoE-over-ATM configuration with a dynamic PPPoE (**pp0**) subscriber interface, create a dynamic profile that defines the **pp0** subscriber interface.

   See "Example: Configuring a Dynamic PPPoE Subscriber Interface over ATM" on page 369.

2. Configure one or more virtual path identifiers (VPIs) on the ATM physical interface.

3. Configure the ATM logical subscriber interface.

   a. Configure the appropriate encapsulation type for your configuration.

      See "ATM for Subscriber Access Encapsulation Types Overview" on page 355.

   b. Configure a virtual circuit identifier (VCI) for each VPI configured on the ATM logical interface.

   c. Configure other interface-specific properties as needed for your configuration.

      See "Guidelines for Configuring ATM for Subscriber Access" on page 356.

4. For static PPPoE-over-ATM configurations, define the static PPPoE (**pp0**) subscriber interface at the **[edit interfaces pp0 unit *logical-unit-number*]** hierarchy level.

   See "Example: Configuring a Static PPPoE Subscriber Interface over ATM" on page 378.

5. (Optional) Verify the configuration for ATM subscriber access.

   See *Verifying and Managing ATM Configurations for Subscriber Access*.

Related Documentation

- ATM for Subscriber Access Overview on page 351
- ATM for Subscriber Access Encapsulation Types Overview on page 355
- Guidelines for Configuring ATM for Subscriber Access on page 356
- Example: Configuring a Dynamic PPPoE Subscriber Interface over ATM on page 369
- Example: Configuring a Static PPPoE Subscriber Interface over ATM on page 378
- Example: Configuring a Static Subscriber Interface for IP Access over ATM on page 387
- Example: Configuring a Static Subscriber Interface for IP Access over Ethernet over ATM on page 393
- Example: Configuring a Static PPP Subscriber Interface over ATM on page 399
- *ATM Interfaces Overview*

CHAPTER 35

Configuring RADIUS Server Options for Subscriber Access

RADIUS Server Options for Subscriber Access

You can specify options that the router uses when communicating with RADIUS authentication and accounting servers for subscriber access.

The following list describes the RADIUS options you can configure:

- **accounting-session-id-format**—The format the router uses to identify the accounting session. The identifier can be in one of the following formats. The router uses **decimal** format by default.
  - **decimal**—For example, **435264**
  - **description**—In the format, **jnpr interface-specifier:subscriber-session-id**. For example, **jnpr fastEthernet 3/2.6:1010101010101**
- **calling-station-id-delimiter**—The character that the router uses as the separator between concatenated values in the Calling-Station-ID string (RADIUS attribute 31).
- **calling-station-id-format**—Optional information that the router includes in the Calling-Station-ID (RADIUS attribute 31).
- **client-accounting-algorithm** and **client-authentication-algorithm**—The method the router uses to access RADIUS accounting and RADIUS authentication servers. You can specify the following methods:
  - **direct**—The default method, in which there is no load balancing. For example, in the direct method, the router always accesses **server1** (the primary server) first, and uses **server2** and **server3** as backup servers.
  - **round-robin**—The method that provides load balancing by rotating router requests among the list of configured RADIUS servers. For example, if three RADIUS servers are configured to support the router, the router sends the first request to **server1**, and

uses **server2** and **server3** as backup servers. The router then sends the second request to **server2**, and uses **server3** and **server1** as backups.

NOTE: When a RADIUS server in the round-robin list becomes unreachable, the next reachable server in the round-robin list is used for the current request. That same server is also used for the next request because it is at the top of the list of available servers. As a result, after a server failure, the server that is used takes up the load of two servers.

- **coa-dynamic-variable-validation**—The optional method that the router uses when processing CoA requests that include changes to a client profile dynamic variable that cannot be applied. The optional configuration specifies that when a CoA operation is unable to apply a requested change to a client profile dynamic variable, subscriber management does not apply any changes to client profile dynamic variables in the CoA request and then responds with a NACK. In the default method, subscriber management does not apply the incorrect update but does apply the other changes to the client profile dynamic variables, and then responds with an ACK message.
- **access-loop-id-local**—The Agent-Remote-Id and Agent-Circuit-Id are generated locally when these values are not present in the client database. The interface description of the logical interface is used as the Agent-Remote-Id and the interface description portion of the NAS-Port-Id using the format **<underlying-interface-name>:<outer-tag>-<inner-tag>** is used as the Agent-Circuit-Id.

NOTE: The NAS-Port-Id format changes (established by **[set access profile *profile-name* radius options interface-description-format]**) are applied before generating the Agent-Circuit-Id.

The NAS-Port-Id format (established by **[set access profile *profile-name* radius options interface-description-format]**) leverages the locally generated Agent-Remote-Id and Agent-Circuit-Id.

- **ethernet-port-type-virtual**—The physical port type of **virtual** that the router uses to authenticate clients. The port type is passed in RADIUS attribute 61 (NAS-Port-Type). By default the router passes a port type of **ethernet** in RADIUS attribute 61.
- **interface-description-format**—The information that is excluded from the interface description that the router passes to RADIUS for inclusion in the RADIUS attribute 87 (NAS-Port-Id). By default, the router includes both the **subinterface** and the **adapter** in the interface description. You can specify:
  - **exclude-adapter**—Exclude the adapter.
  - **exclude-subinterface**—Exclude the subinterface.
- **nas-identifier**—The value for the client RADIUS attribute 32 (NAS-Identifier), which is used for authentication and accounting requests. You can specify a string in the range 1 through 64 characters.

- **nas-port-extended-format**—The extended format for RADIUS attribute 5 (NAS-Port) and for the width of the fields in the NAS-Port attribute that the RADIUS client uses. You can specify:
  - **adapter-width** ***width***—Number of bits in the adapter field.
  - **port-width** ***width***—Number of bits in the port field.
  - **slot-width** ***width***—Number of bits in the slot field.
  - **stacked-vlan-width** ***width***—Number of bits in the SVLAN ID field.
  - **vlan-width** ***width***—Number of bits in the VLAN ID field.

  NOTE: **The total of the widths must not exceed 32 bits, or the configuration will fail.**

  You can configure an extended format for the NAS-Port attribute for both Ethernet subscribers and ATM subscribers. For ATM subscribers, you can specify:
  - **adapter-width**—Number of bits in the ATM adapter field, in the range 0 through 32
  - **port-width**—Number of bits in the ATM port field, in the range 0 through 32
  - **slot-width**—Number of bits in the ATM slot field, in the range 0 through 32
  - **vpi-width**—Number of bits in the ATM virtual path identifier (VPI) field, in the range 0 through 32
  - **vci-width**—Number of bits in the ATM virtual circuit identifier (VCI) field, in the range 0 through 32

  NOTE: **For ATM subscribers, the combined total of the widths of all fields must not exceed 32 bits, or the configuration fails. The router may truncate the values of individual fields depending on the bit width you specify.**
- **nas-port-id-delimiter**—The character used as the separator between values in the NAS-Port-ID string.
- **nas-port-id-format**—Optional information included in RADIUS attribute 87 (NAS-Port-ID).
- **nas-port-type**—The port type used to authenticate subscribers.
- **revert-interval**—The number of seconds that the router waits after a server has become unreachable. The router rechecks the connection to the server when the **revert-interval** expires. If the server is then reachable, it is used in accordance with the order of the server list. You can configure from 0 (off) through 604800 seconds. The default is 60 seconds.
- **vlan-nas-port-stacked-format**—The format that turns off RADIUS attribute 5 (NAS-Port) to include the S-VLAN ID, in addition to the VLAN ID, for subscribers on Ethernet interfaces.

Related Documentation

- Configuring RADIUS Server Options for Subscriber Access on page 364

Configuring RADIUS Server Options for Subscriber Access

You can specify options that the router or switch uses when communicating with RADIUS authentication and accounting servers for subscriber access.

To configure RADIUS authentication and accounting server options:

1. Specify that you want to configure RADIUS.

   [edit access profile isp-bos-metro-fiber-basic]
   user@host# **edit radius**

2. Specify that you want to configure RADIUS options.

   [edit access profile isp-bos-metro-fiber-basic radius]
   user@host# **edit options**

3. (Optional) Configure the method the router or switch uses to access RADIUS accounting servers.

   [edit access profile isp-bos-metro-fiber-basic radius options]
   user@host# **set client-accounting-algorithm round-robin**

4. (Optional) Configure the method the router or switch uses to access RADIUS authentication servers.

   [edit access profile isp-bos-metro-fiber-basic radius options]
   user@host# **set client-authentication-algorithm round-robin**

5. (Optional) Configure the format the router or switch uses to identify the accounting session.

   [edit access profile isp-bos-metro-fiber-basic radius options]
   user@host# **set accounting-session-id-format decimal**

6. (Optional) Specify that the Agent-Remote-Id and Agent-Circuit-Id are generated locally when these values are not present in the client database.

   [edit access profile isp-bos-metro-fiber-basic radius options]
   user@host# **set access-loop-id-local**

7. (Optional) Specify the information that is excluded from the interface description that the router or switch passes to RADIUS for inclusion in RADIUS attribute 87 (NAS-Port-Id).

   [edit access profile isp-bos-metro-fiber-basic radius options]
   user@host# **set interface-description-format exclude-adapter**

8. (Optional) Configure the value for the client RADIUS attribute 32 (NAS-Identifier), which is used for authentication and accounting requests.

   [edit access profile isp-bos-metro-fiber-basic radius options]
   user@host# **set nas-identifier 56**

9. (Optional) Configure the RADIUS client to use the extended format for RADIUS attribute 5 (NAS-Port) and specify the width of the fields in the NAS-Port attribute. The total of the widths must not exceed 32 bits, or the configuration fails.

   - For Ethernet subscribers:

     [edit access profile isp-bos-metro-fiber-basic radius options]
     user@host# **set** nas-port-extended-format **ae-width 10 slot-width 4 adapter-width 2 port-width 4 stacked-vlan-width 10 vlan-width 2**

   - For ATM subscribers:

     [edit access profile retailer01 radius options]
     user@host# **set** nas-port-extended-format **atm slot-width 3 adapter-width 2 port-width 3 vpi-width 8 vci-width 16**

10. (Optional) Configure the delimiter character that the router inserts between values in RADIUS attribute 87 (NAS-Port-ID).

    [edit access profile isp-bos-metro-fiber-basic radius options]
    user@host# **set nas-port-id-delimiter %**

11. (Optional) Configure the information that the router includes in RADIUS attribute 87 (NAS-Port-ID).

    [edit access profile isp-bos-metro-fiber-basic radius options]
    user@host# **set nas-port-id-format agent-circuit-id agent-remote-id**

12. (Optional) Configure the delimiter character that the router inserts between values in RADIUS attribute 31 (Calling-Station-ID).

    [edit access profile isp-bos-metro-fiber-basic radius options]
    user@host# **set calling-station-id-delimiter "%"**

13. (Optional) Configure the information that the router includes in RADIUS attribute 31 (Calling-Station-ID).

    [edit access profile isp-bos-metro-fiber-basic radius options]
    user@host# **set calling-station-id-format agent-circuit-id agent-remote-id**

14. (Optional) Configure the port type that is included in RADIUS attribute 61 (NAS-Port-Type). This specifies the port type the router uses to authenticate subscribers.

    [edit access profile isp-bos-metro-fiber-basic radius options]
    user@host# **set nas-port-type ethernet wireless-ieee80211**

    NOTE: This statement is ignored if you configure the **ethernet-port-type-virtual** in the same access profile.

15. (Optional) Configure the router or switch to use a port type of **virtual** to authenticate clients.

    [edit access profile isp-bos-metro-fiber-basic radius options]
    user@host# **set ethernet-port-type-virtual**

NOTE: This statement takes precedence over the **nas-port-type** statement if you include both in the same access profile.

16. (Optional) Configure the number of seconds that the router or switch waits after a server has become unreachable.

    ```
    [edit access profile isp-bos-metro-fiber-basic radius options]
    user@host# set revert-interval 259200
    ```

17. (Optional) Specify that RADIUS attribute 5 (NAS-Port) includes the S-VLAN ID, in addition to the VLAN ID, for subscribers on Ethernet interfaces.

    ```
    [edit access profile isp-bos-metro-fiber-basic radius options]
    user@host# set vlan-nas-port-stacked-format
    ```

18. (Optional) Configure the router to use the optional behavior when processing CoA requests that include changes to client profile dynamic variables.

    ```
    [edit access profile isp-bos-metro-fiber-basic radius options]
    user@host# set coa-dynamic-variable-validation
    ```

Related Documentation

- RADIUS Server Options for Subscriber Access on page 361
- *Configuring Router or Switch Interaction with RADIUS Servers*
- *Manual Configuration of the NAS-Port-Type RADIUS Attribute*
- *Configuring a NAS-Port-ID with Additional Options*
- *Configuring a Calling-Station-ID with Additional Attributes*
- *Example: Configuring RADIUS-Based Subscriber Authentication and Accounting*

Configuring the RADIUS NAS-Port Extended Format for ATM Interfaces

As an alternative to globally configuring an extended format for the NAS-Port (5) RADIUS attribute in an access profile, you can configure the NAS-Port extended format on a per-physical interface basis for both Ethernet subscribers and ATM subscribers as part of a NAS-Port options definition. The NAS-Port extended format configures the number of bits (bit width) in each field of the NAS-Port attribute, including: slot, adapter, port, ATM virtual path identifier (VPI), and ATM virtual circuit identifier (VCI).

To configure the NAS-Port extended format for an ATM interface, include one or both of the following options in the **nas-port-extended-format** statement along with the other options as appropriate for your needs:

- **vpi-width**—Number of bits in the ATM VPI field, in the range 1 through 32
- **vci-width**—Number of bits in the ATM VCI field, in the range 1 through 32

NOTE: **For ATM subscribers, the combined total of the widths of all fields must not exceed 32 bits, or the configuration fails. The router may truncate the values of individual fields depending on the bit width you specify.**

To configure an extended format for the NAS-Port RADIUS attribute for an ATM interface:

1. Specify the ATM interface you want to configure.

   [edit]
   user@host# **edit interfaces *interface-name***

2. Specify that you want to configure RADIUS options for a physical interface.

   [edit interfaces *interface-name*]
   user@host# **edit radius-options**

3. Create a named NAS-Port options definition.

   [edit interfaces *interface-name* radius-options]
   user@host# **edit nas-port-options *nas-port-options-name***

4. Configure the NAS-Port extended format.

   [edit interfaces *interface-name* radius-options nas-port-options *nas-port-options-name*]
   user@host# **set nas-port-extended-format slot-width *width* adapter-width *width* port-width *width* vpi-width *width* vci-width *width***

The following example shows a NAS-Port options definition named boston-subscribers for ATM interface at-1/0/4 that configures a NAS-Port extended format with an ATM slot width of 6 bits, ATM adapter width of 3 bits, ATM port width of 4 bits, ATM VPI width of 12 bits, and ATM VCI width of 24 bits.

```
[edit interfaces at-1/0/4 radius-options]
nas-port-options boston-subscribers {
  nas-port-extended-format {
    slot-width 6;
```

```
        adapter-width 3;
        port-width 4;
        vpi-width 12;
        vci-width 24;
    }
}
```

Related Documentation

- *Configuring RADIUS NAS-Port Options for Subscriber Access per Physical Interface, VLAN, or Stacked VLAN*
- RADIUS Server Options for Subscriber Access on page 361
- Configuring RADIUS Server Options for Subscriber Access on page 364

CHAPTER 36

Configuring PPPoE Subscriber Interfaces Over ATM

Example: Configuring a Dynamic PPPoE Subscriber Interface over ATM

This example illustrates a Point-to-Point Protocol over Ethernet (PPPoE) over ATM configuration that creates a dynamic PPPoE (**pp0**) subscriber interface over a static ATM underlying interface on an MX Series router. The router must have Module Port Concentrator/Modular Interface Card (MPC/MIC) interfaces that use an ATM MIC with small form-factor pluggable transceiver (SFP).

NOTE: You can also configure a *static* PPPoE interface over a static ATM underlying interface on an MX Series router with an ATM MIC with SFP installed. For information, see "Example: Configuring a Static PPPoE Subscriber Interface over ATM" on page 378.

Requirements

This example uses the following software and hardware components:

- MX Series 3D Universal Edge Router
- ATM MIC with SFP (Model Number MIC-3D-8OC3-2OC12-ATM) and compatible MPC1 or MPC2

Before you begin:

1. Make sure the MX Series router you are using has an ATM MIC with SFP installed and operational.

- For information about compatible MPCs for the ATM MIC with SFP, see the *MX Series Interface Module Reference*.
- For information about installing MPCs and MICs in an MX Series router, see the *Hardware Guide* for your MX Series router model.

2. Make sure you understand how to configure and use static ATM interfaces.

   See *ATM Interfaces Overview*.

3. Make sure you understand how to configure and use dynamic PPPoE subscriber interfaces.

   - For overview information, see "Subscriber Interfaces and PPPoE Overview" on page 141
   - For configuration instructions, see "Configuring Dynamic PPPoE Subscriber Interfaces Using Dynamic Profiles" on page 145

Overview

By using the ATM MIC with SFP and a supported MPC, you can configure an MX Series router to support dynamic PPPoE subscriber access over an ATM network. PPPoE-over-ATM configurations on MX Series routers consist of one or more dynamically created PPPoE (**pp0**) subscriber interfaces over a static ATM underlying interface. Most PPPoE and subscriber services features supported on terminated connections and tunneled (L2TP access concentrator, or LAC) connections are also supported for PPPoE-over-ATM connections on an MX Series router.

Optionally, you can dynamically apply subscriber services such as class of service (CoS) and firewall filters to the PPPoE subscriber interface by configuring these services in the dynamic profile that creates the **pp0** subscriber interface. In this example, the PPPoE dynamic profile (pppoe-profile) applies CoS traffic shaping parameters to the dynamic **pp0** subscriber interface. Configuring CoS and firewall filters in this manner enables you to efficiently and economically provide these services to PPPoE subscribers accessing the router over an ATM network using ATM Adaptation Layer 5 (AAL5) permanent virtual connections (PVCs).

This example includes the following basic steps to configure dynamic PPPoE-over-ATM subscriber access on an MX Series router:

1. Create a PPPoE dynamic profile named pppoe-profile for the **pp0** subscriber interface that includes all of the following:

   - The logical unit number, represented by the **$junos-interface-unit** predefined dynamic variable
   - The name of the underlying ATM interface, represented by the **$junos-underlying-interface** predefined dynamic variable
   - The **server** statement, which configures the router to act as a PPPoE server

   NOTE: Configuring the router to act as a PPPoE client is not supported.

- The unnumbered address (lo0.0) for the IPv4 (**inet**) protocol family
- CoS traffic shaping parameters

2. Statically configure the ATM physical interface at-1/0/0 with virtual path identifier (VPI) 3.
3. Statically configure logical unit 2 on the ATM physical interface (at-1/0/0.2) with at least the following properties:
   - PPPoE-over-ATM logical link control (LLC) encapsulation (**ppp-over-ether-over-atm-llc**)
   - Virtual circuit identifier (VCI) 2 on VPI 3. The combination of VPIs and VCIs provisions the ATM AAL5 PVC for access over the ATM network.
   - PPPoE-specific options at the **[edit interfaces** ***interface-name*** **unit** ***logical-unit-number*** **family pppoe]** hierarchy level, including at least the name of the associated PPPoE dynamic profile (pppoe-profile) that creates the pp0 dynamic subscriber interface

     In dynamic PPPoE-over-ATM configurations, each **pp0** interface defined in the dynamic profile corresponds to a dynamic PPPoE subscriber interface.

NOTE: For dynamic or static PPPoE-over-ATM configurations on MX Series routers, You must specify PPPoE-specific options in the family pppoe stanza at the [edit interfaces *interface-name* unit *logical-unit-number*] hierarchy level. Specifying PPPoE-specific options in the pppoe-underlying-options stanza at the [edit interfaces *interface-name* unit *logical-unit-number*] hierarchy level is not supported for these configurations.

Configuration

To configure a dynamic PPPoE subscriber interface over an underlying ATM interface, perform these tasks:

- Configuring the PPPoE Dynamic Profile on page 372
- Configuring the ATM Physical Interface on page 374
- Configuring the Dynamic PPPoE Subscriber Interface on Logical Unit 2 on page 375

CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them in 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.

```
# PPPoE Dynamic Profile
set dynamic-profiles pppoe-profile interfaces pp0 unit "$junos-interface-unit" ppp-options
  chap
set dynamic-profiles pppoe-profile interfaces pp0 unit "$junos-interface-unit"
  pppoe-options underlying-interface "$junos-underlying-interface"
set dynamic-profiles pppoe-profile interfaces pp0 unit "$junos-interface-unit"
  pppoe-options server
```

```
set dynamic-profiles pppoe-profile interfaces pp0 unit "$junos-interface-unit"
  no-keepalives
set dynamic-profiles pppoe-profile interfaces pp0 unit "$junos-interface-unit" family
  inet unnumbered-address lo0.0
set dynamic-profiles pppoe-profile class-of-service traffic-control-profiles tcp-test
  shaping-rate 10m
set dynamic-profiles pppoe-profile class-of-service interfaces pp0 unit
  "$junos-interface-unit" output-traffic-control-profile tcp-test
#
# ATM Physical Interface
set interfaces at-1/0/0 atm-options vpi 3
#
# Logical Unit 2
set interfaces at-1/0/0 atm-options vpi 3
set interfaces at-1/0/0 unit 2 encapsulation ppp-over-ether-over-atm-llc
set interfaces at-1/0/0 unit 2 vci 3.2
set interfaces at-1/0/0 unit 2 family pppoe access-concentrator ac-pppoeoa
set interfaces at-1/0/0 unit 2 family pppoe duplicate-protection
set interfaces at-1/0/0 unit 2 family pppoe dynamic-profile pppoe-profile
set interfaces at-1/0/0 unit 2 family pppoe max-sessions 3
set interfaces at-1/0/0 unit 2 family pppoe short-cycle-protection
```

Configuring the PPPoE Dynamic Profile

Step-by-Step Procedure

To configure the PPPoE dynamic profile for the **pp0** subscriber interface:

1. Name the dynamic profile.

   ```
   [edit]
   user@host# edit dynamic-profiles pppoe-profile
   ```

2. Specify that you want to configure the **pp0** (PPPoE) interface.

   ```
   [edit dynamic-profiles pppoe-profile]
   user@host# edit interfaces pp0
   ```

3. Specify that you want to configure the logical unit represented by the **$junos-interface-unit** predefined variable.

   ```
   [edit dynamic-profiles pppoe-profile interfaces pp0]
   user@host# edit unit $junos-interface-unit
   ```

   The **$junos-interface-unit** variable is dynamically replaced with the actual unit number supplied by the network when the subscriber logs in.

4. Configure PPPoE-specific options for the **pp0** interface.

   a. Configure the ATM underlying interface represented by the **$junos-underlying-interface** predefined variable.

   ```
   [edit dynamic-profiles pppoe-profile interfaces pp0 unit "$junos-interface-unit"]
   user@host# set pppoe-options underlying-interface $junos-underlying-interface
   ```

   The **$junos-underlying-interface** variable is dynamically replaced with the actual name of the underlying interface supplied by the network when the subscriber logs in.

   b. Configure the router to act as a PPPoE server, also known as a remote access concentrator.

[edit dynamic-profiles pppoe-profile interfaces pp0 unit "$junos-interface-unit"]
user@host# **set** pppoe-options server

5. Configure Challenge Handshake Authentication Protocol (CHAP) authentication for the **pp0** interface.

   [edit dynamic-profiles pppoe-profile interfaces pp0 unit "$junos-interface-unit"]
   user@host# **set** ppp-options chap

6. Disable sending keepalive messages on the interface.

   [edit dynamic-profiles pppoe-profile interfaces pp0 unit "$junos-interface-unit"]
   user@host# **set** no-keepalives

7. Configure the protocol family for the **pp0** interface.

   a. Specify that you want to configure the IPv4 (**inet**) protocol family.

      [edit dynamic-profiles pppoe-profile interfaces pp0 unit "$junos-interface-unit"]
      user@host# **edit** family **inet**

   b. Configure the unnumbered address for the protocol family.

      [edit dynamic-profiles pppoe-profile interfaces pp0 unit "$junos-interface-unit" family inet]
      user@host# **set** unnumbered-address **lo0.0**
      user@host# **up 4**

8. Configure CoS traffic shaping parameters in the dynamic profile for the pp0 subscriber interface.

   a. Specify that you want to configure CoS traffic shaping parameters.

      [edit dynamic-profiles pppoe-profile]
      user@host# **edit** class-of-service

   b. Create a traffic-control profile.

      [edit dynamic-profiles pppoe-profile class-of-service]
      user@host# **edit** traffic-control-profiles **tcp-test**

   c. Configure the traffic shaping rate.

      [edit dynamic-profiles pppoe-profile class-of-service traffic-control-profiles tcp-test]
      user@host# **set shaping-rate 10m**
      user@host# **up 2**

   d. Apply the traffic shaping parameters to the pp0 dynamic subscriber interface.

      [edit dynamic-profiles pppoe-profile class-of-service]
      user@host# **edit** interfaces pp0 unit **$junos-interface-unit**

   e. Apply the output traffic scheduling and shaping profile to the interface.

      [edit dynamic-profiles pppoe-profile class-of-service interfaces pp0 unit "$junos-interface-unit]
      user@host# **set** output-traffic-control-profile **tcp-test**

Results From the **[edit]** hierarchy level in configuration mode, confirm the results of the PPPoE dynamic profile configuration by issuing the **show dynamic-profiles pppoe-profile**

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

```
[edit]
user@host# show dynamic-profiles pppoe-profile
interfaces {
  pp0 {
    unit "$junos-interface-unit" {
      ppp-options {
        chap;
      }
      pppoe-options {
        underlying-interface "$junos-underlying-interface";
        server;
      }
      no-keepalives;
      family inet {
        unnumbered-address lo0.0;
      }
    }
  }
}
class-of-service {
  traffic-control-profiles {
    tcp-test {
      shaping-rate 10m;
    }
  }
  interfaces {
    pp0 {
      unit "$junos-interface-unit" {
        output-traffic-control-profile tcp-test;
      }
    }
  }
}
```

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

Configuring the ATM Physical Interface

Step-by-Step Procedure

To configure the ATM physical interface:

1. Specify that you want to configure ATM-specific options on the physical interface.

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

2. Configure one or more VPIs on the physical interface.

```
[edit interfaces at-1/0/0 atm-options]
user@host# set vpi 3
```

Results

From the **[edit]** hierarchy level in configuration mode, confirm the results of the ATM physical interface configuration by issuing the **show interfaces at-1/0/0** command. If the output does not display the intended configuration, repeat the instructions in this example to correct it.

```
[edit]
user@host# show interfaces at-1/0/0
atm-options {
  vpi 3;
}
```

If you are done configuring the ATM physical interface, enter **commit** from configuration mode.

Configuring the Dynamic PPPoE Subscriber Interface on Logical Unit 2

Step-by-Step Procedure

To configure the dynamic PPPoE subscriber interface on logical unit 2:

1. Configure PPPoE-over-ATM LLC encapsulation on the interface.

   ```
   [edit interfaces at-1/0/0 unit 2]
   user@host# set encapsulation ppp-over-ether-over-atm-llc
   ```

2. Configure the VCI for the logical interface.

   ```
   [edit interfaces at-1/0/0 unit 2]
   user@host# set vci 3.2
   ```

   This statement configures VCI 2 on VPI 3.

3. Specify that you want to configure the PPPoE protocol family.

   ```
   [edit interfaces at-1/0/0 unit 2]
   user@host# edit family pppoe
   ```

4. Associate the interface with the dynamic profile that creates the dynamic PPPoE subscriber interface.

   ```
   [edit interfaces at-1/0/0 unit 2 family pppoe]
   user@host# set dynamic-profile pppoe-profile
   ```

5. Configure additional PPPoE-specific options for the dynamic subscriber interface.

   ```
   [edit interfaces at-1/0/0 unit 2 family pppoe]
   user@host# set max-sessions 3
   user@host# set duplicate-protection
   user@host# set short-cycle-protection
   user@host# set access-concentrator ac-pppoeoa
   ```

Results

From the **[edit]** hierarchy level in configuration mode, confirm the results of the dynamic PPPoE subscriber interface configuration on logical unit 2 by issuing the **show interfaces at-1/0/0.2** command. If the output does not display the intended configuration, repeat the instructions in this example to correct it.

```
[edit]
user@host# show interfaces at-1/0/0.2
encapsulation ppp-over-ether-over-atm-llc;
vci 3.2;
family pppoe {
  access-concentrator ac-pppoeoa;
```

```
    duplicate-protection;
    dynamic-profile pppoe-profile;
    max-sessions 3;
    short-cycle-protection;
  }
```

If you are done configuring the dynamic PPPoE subscriber interface on logical unit 2, enter **commit** from configuration mode.

Verification

To confirm that the dynamic PPPoE subscriber interface is properly configured on ATM interface at-1/0/0.2, perform the following tasks:

Verifying the ATM Physical Interface Configuration

Purpose Verify that ATM physical interface at-1/0/0 is properly configured for use with ATM PVCs.

Action From operational mode, issue the **show interfaces at-1/0/0** command.

For brevity, this **show** command output includes only the configuration that is relevant to the at-1/0/0 physical interface. Any other configuration on the system has been replaced with ellipses (...).

```
user@host> show interfaces at-1/0/0
Physical interface: at-1/0/0, Enabled, Physical link is Up
  Interface index: 173, SNMP ifIndex: 592
  Link-level type: ATM-PVC, MTU: 2048, Clocking: Internal, SDH mode, Speed: OC3,
 Loopback: None, Payload scrambler: Enabled
  Device flags   : Present Running
  Link flags     : None
  CoS queues     : 8 supported, 8 maximum usable queues
  Schedulers     : 0
  Current address: 00:1f:12:bc:4a:95
  Last flapped   : 2012-09-17 07:21:19 PDT (08:26:16 ago)
  Input rate     : 0 bps (0 pps)
  Output rate    : 0 bps (0 pps)
  SDH   alarms   : None
  SDH   defects  : None
    VPI 3
      Flags: Active
      Total down time: 0 sec, Last down: Never
  Traffic statistics:
         Input  packets:                   0
         Output packets:                   0
...
```

Meaning **ATM-PVC** in the Link-level Type field indicates that encapsulation for ATM permanent virtual circuits is being used on ATM physical interface at-1/0/0. The **Active** flag for VPI 3 indicates that the virtual path is up and operational.

Verifying the Dynamic PPPoE Subscriber Interface Configuration on Logical Unit 2

Purpose Verify that the dynamic PPPoE subscriber interface is properly configured on logical unit 2 (at-1/0/0.2).

Action From operational mode, issue the **show interfaces at-1/0/0.2** command.

```
user@host> show interfaces at-1/0/0.2
  Logical interface at-1/0/0.2 (Index 350) (SNMP ifIndex 1701)
    Flags: Point-To-Point SNMP-Traps 0x4000 Encapsulation: PPPoE-over-ATM-LLC
    Input packets : 0
    Output packets: 0
    Protocol pppoe
      Dynamic Profile: pppoe-profile,
      Service Name Table: None,
      Max Sessions: 3, Max Sessions VSA Ignore: Off,
      Duplicate Protection: On, Short Cycle Protection: mac-address,
      AC Name: ac-pppoeoa
    VCI 3.2
      Flags: Active
      Total down time: 0 sec, Last down: Never
        Input packets : 0
        Output packets: 0
```

Meaning **PPPoE-over-ATM-LLC** in the Encapsulation field indicates that logical interface at-1/0/0.2 is properly configured for PPPoE-over-ATM LLC encapsulation. **Protocol pppoe** indicates that the PPPoE protocol family has been properly configured on the logical interface. The Dynamic Profile field indicates that dynamic profile **pppoe-profile** creates the dynamic PPPoE subscriber interface. The **Active** flag for VCI 3.2 indicates that VCI 2 on VPI 3 is up and operational.

Verifying the PPPoE Underlying Interface Configuration

Purpose Verify that the underlying interface is properly configured for dynamic PPPoE-over-ATM subscriber access.

Action From operational mode, issue the **show pppoe underlying-interfaces at-1/0/0.2** command.

```
user@host> show pppoe underlying-interfaces at-1/0/0.2 detail
at-1/0/0.2 Index 350
  State: Static, Dynamic Profile: pppoe-profile,
  Max Sessions: 3, Max Sessions VSA Ignore: Off,
  Active Sessions: 0,
  Service Name Table: None,
  Duplicate Protection: On, Short Cycle Protection: mac-address,
  AC Name: ac-pppoeoa,
```

Meaning This command indicates that ATM logical interface at-1/0/0.2 is properly configured as the PPPoE underlying interface. **Static** in the State field indicates that at-1/0/0/.2 is statically configured. The Dynamic Profile field indicates that **pppoe-profile** is the name of the dynamic profile used to create this interface. The remaining fields display information about the PPPoE-specific interface options configured for the PPPoE underlying interface at the **[edit interfaces at-1/0/0 unit 2 family pppoe]** hierarchy level.

Related Documentation

- ATM for Subscriber Access Overview on page 351
- Configuring ATM for Subscriber Access on page 358
- Example: Configuring a Static PPPoE Subscriber Interface over ATM on page 378
- Example: Configuring a Static Subscriber Interface for IP Access over ATM on page 387
- Example: Configuring a Static Subscriber Interface for IP Access over Ethernet over ATM on page 393
- Example: Configuring a Static PPP Subscriber Interface over ATM on page 399

Example: Configuring a Static PPPoE Subscriber Interface over ATM

This example illustrates a Point-to-Point Protocol over Ethernet (PPPoE) over ATM configuration that creates a static PPPoE (**pp0**) subscriber interface over a static ATM underlying interface on an MX Series router. The router must have Module Port Concentrator/Modular Interface Card (MPC/MIC) interfaces that use an ATM MIC with small form-factor pluggable transceiver (SFP).

NOTE: You can also configure a *dynamic* PPPoE interface over a static ATM underlying interface on an MX Series router with an ATM MIC with SFP installed. For information, see "Example: Configuring a Dynamic PPPoE Subscriber Interface over ATM" on page 369.

- Requirements on page 378
- Overview on page 379
- Configuration on page 380
- Verification on page 383

Requirements

This example uses the following software and hardware components:

- MX Series 3D Universal Edge Router
- ATM MIC with SFP (Model Number MIC-3D-8OC3-2OC12-ATM) and compatible MPC1 or MPC2

Before you begin:

1. Make sure the MX Series router you are using has an ATM MIC with SFP installed and operational.
   - For information about compatible MPCs for the ATM MIC with SFP, see the *MX Series Interface Module Reference*.
   - For information about installing MPCs and MICs in an MX Series router, see the *Hardware Guide* for your MX Series router model.

2. Make sure you understand how to configure and use static ATM interfaces.

   See *ATM Interfaces Overview*.

Overview

By using the ATM MIC with SFP and a supported MPC, you can configure an MX Series router to support static PPPoE subscriber access over an ATM network using ATM Adaptation Layer 5 (AAL5) permanent virtual connections (PVCs). PPPoE-over-ATM configurations on MX Series routers consist of one or more statically created PPPoE (**pp0**) logical subscriber interfaces over a static ATM underlying interface. Most PPPoE and subscriber services features supported on terminated connections and tunneled (L2TP access concentrator, or LAC) connections are also supported for PPPoE-over-ATM connections on an MX Series router.

This example include the following basic steps to configure static PPPoE-over-ATM subscriber access on an MX Series router:

1. Statically configure ATM physical interface at-1/0/6 with virtual path identifier (VPI) 6.
2. Statically configure logical unit 2 on the ATM physical interface (at-1/0/6.2) with the following properties:
   - PPPoE-over-ATM logical link control (LLC) encapsulation (**ppp-over-ether-over-atm-llc**)
   - Virtual circuit identifier (VCI) 2 on VPI 6. The combination of VPIs and VCIs provisions the ATM AAL5 PVC for access over the ATM network.
   - (Optional) PPPoE-specific options at the **[edit interfaces *interface-name* unit *logical-unit-number* family pppoe]** hierarchy level

   NOTE: For dynamic or static PPPoE-over-ATM configurations on MX Series routers, You must specify PPPoE-specific options in the **family pppoe** stanza at the **[edit interfaces *interface-name* unit *logical-unit-number*]** hierarchy level. Specifying PPPoE-specific options in the **pppoe-underlying-options** stanza at the **[edit interfaces *interface-name* unit *logical-unit-number*]** hierarchy level is not supported for these configurations.

3. Statically configure the **pp0** logical subscriber interface (pp0.2) with at least the following properties:
   - The name of the underlying ATM interface (at-1/0/6.2)
   - The **server** statement, which configures the router to act as a PPPoE server
   - The unnumbered address (lo0.0) for the **inet** (IPv4) or **inet6** (IPv6) protocol family

   In static PPPoE-over-ATM configurations, each **pp0** logical interface configured at the **[edit interfaces pp0 unit *logical-unit-number*]** hierarchy level corresponds to a static PPPoE subscriber interface.

Configuration

To configure a static PPPoE subscriber interface over an underlying ATM interface, perform these tasks:

- Configuring the ATM Physical Interface on page 380
- Configuring Encapsulation, VCI, and PPPoE Options on Logical Unit 2 on page 381
- Configuring the Static PPPoE Subscriber Interface on page 382

CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them in 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.

```
# ATM Physical Interface
set interfaces at-1/0/6 atm-options vpi 6
#
# Logical Unit 2
set interfaces at-1/0/6 unit 2 encapsulation ppp-over-ether-over-atm-llc
set interfaces at-1/0/6 unit 2 vci 6.2
set interfaces at-1/0/6 unit 2 family pppoe access-concentrator ac-pppoeoa
set interfaces at-1/0/6 unit 2 family pppoe duplicate-protection
set interfaces at-1/0/6 unit 2 family pppoe max-sessions 3
set interfaces at-1/0/6 unit 2 family pppoe max-sessions-vsa-ignore
set interfaces at-1/0/6 unit 2 family pppoe short-cycle-protection lockout-time-min 120
set interfaces at-1/0/6 unit 2 family pppoe short-cycle-protection lockout-time-max 240
#
# Static PPPoE Subscriber Interface
set interfaces pp0 unit 2 ppp-options chap
set interfaces pp0 unit 2 pppoe-options underlying-interface at-1/0/6.2
set interfaces pp0 unit 2 pppoe-options server
set interfaces pp0 unit 2 keepalives interval 10
set interfaces pp0 unit 2 family inet unnumbered-address lo0.0
```

Configuring the ATM Physical Interface

Step-by-Step Procedure

To configure the ATM physical interface:

1. Specify that you want to configure ATM-specific options on the physical interface.

   ```
   [edit interfaces at-1/0/6]
   user@host# edit atm-options
   ```

2. Configure one or more VPIs on the physical interface.

   ```
   [edit interfaces at-1/0/6 atm-options]
   user@host# set vpi 6
   ```

Results

From the **[edit]** hierarchy level in configuration mode, confirm the results of the ATM physical interface configuration by issuing the **show interfaces at-1/0/6** command. If the output does not display the intended configuration, repeat the instructions in this example to correct it.

```
[edit]
```



```
user@host# show interfaces at-1/0/6
atm-options {
   vpi 6;
}
```

If you are done configuring the ATM physical interface, enter **commit** from configuration mode.

Configuring Encapsulation, VCI, and PPPoE Options on Logical Unit 2

Step-by-Step Procedure

To configure encapsulation, VCI, and PPPoE options on logical unit 2:

1. Configure PPPoE-over-ATM LLC encapsulation on the interface.

   ```
   [edit interfaces at-1/0/6 unit 2]
   user@host# set encapsulation ppp-over-ether-over-atm-llc
   ```

2. Configure the VCI for the logical interface.

   ```
   [edit interfaces at-1/0/6 unit 2]
   user@host# set vci 6.2
   ```

   This statement configures VCI 2 on VPI 6.

3. Specify that you want to configure the PPPoE protocol family.

   ```
   [edit interfaces at-1/0/6 unit 2]
   user@host# edit family pppoe
   ```

4. Configure additional PPPoE-specific options for the dynamic subscriber interface.

   ```
   [edit interfaces at-1/0/6 unit 2 family pppoe]
   user@host# set duplicate-protection
   user@host# set short-cycle-protection lockout-time-min 120 lockout-time-max
     240
   user@host# set max-sessions 3
   user@host# set max-sessions-vsa-ignore
   user@host# set access-concentrator ac-pppoeoa
   ```

Results

From the **[edit]** hierarchy level in configuration mode, confirm the results of the configuration on logical unit 2 by issuing the **show interfaces at-1/0/6.2** command. If the output does not display the intended configuration, repeat the instructions in this example to correct it.

```
[edit]
user@host# show interfaces at-1/0/6.2
encapsulation ppp-over-ether-over-atm-llc;
vci 6.2;
family pppoe {
   access-concentrator ac-pppoeoa;
   duplicate-protection;
   max-sessions 3;
   max-sessions-vsa-ignore;
   short-cycle-protection {
      lockout-time-min 120;
      lockout-time-max 240;
   }
}
```


If you are done configuring logical unit 2, enter **commit** from configuration mode.

Configuring the Static PPPoE Subscriber Interface

Step-by-Step Procedure

To configure the static PPPoE subscriber interface:

1. Specify that you want to configure the **pp0** subscriber interface on logical unit 2.

   [edit]
   user@host# **edit interfaces pp0 unit 2**

2. Specify that you want to configure PPP options for the subscriber interface.

   [edit interfaces pp0 unit 2]
   user@host# **edit ppp-options**

3. Configure Challenge Handshake Authentication Protocol (CHAP) authentication for the subscriber interface.

   [edit interfaces pp0 unit 2 ppp-options]
   user@host# **set chap**
   user@host# **up**

4. Specify that you want to configure PPPoE-specific options.

   [edit interfaces pp0 unit 2]
   user@host# **edit pppoe-options**

5. Associate the PPPoE subscriber interface with the underlying ATM interface.

   [edit interfaces pp0 unit 2 pppoe-options]
   user@host# **set underlying-interface at-1/0/6.2**

6. Configure the router to act as a PPPoE server, also known as a remote access concentrator.

   [edit interfaces pp0 unit 2 pppoe-options]
   user@host# **set server**
   user@host# **up**

7. Configure the interval for sending keepalive requests.

   [edit interfaces pp0 unit 2]
   user@host# **set keepalives interval 10**

8. Specify that you want to configure the IPv4 (**inet**) protocol family.

   [edit interfaces pp0 unit 2]
   user@host# **edit family inet**

9. Configure the unnumbered address for the protocol family.

   [edit interfaces pp0 unit 2 family inet]
   user@host# **set unnumbered-address lo0.0**

Results

From the **[edit]** hierarchy level in configuration mode, confirm the results of the static PPPoE subscriber interface configuration by issuing the **show interfaces pp0** command. If the output does not display the intended configuration, repeat the instructions in this example to correct it.

```
[edit]
user@host# show interfaces pp0
unit 2 {
  ppp-options {
    chap;
  }
  pppoe-options {
    underlying-interface at-1/0/6.2;
    server;
  }
  keepalives interval 10;
  family inet {
    unnumbered-address lo0.0;
  }
}
```

If you are done configuring the static PPPoE subscriber interface, enter **commit** from configuration mode.

Verification

To confirm that the static PPPoE subscriber interface pp0.2 is properly configured on ATM underlying interface at-1/0/6.2, perform the following tasks:

- Verifying the ATM Physical Interface Configuration on page 383
- Verifying the Encapsulation, VCI, and PPPoE Options Configuration on Logical Unit 2 on page 384
- Verifying the Static PPPoE Subscriber Interface Configuration on page 384
- Verifying the PPPoE Underlying Interface Configuration on page 385

Verifying the ATM Physical Interface Configuration

Purpose Verify that ATM physical interface at-1/0/6 is properly configured for use with ATM PVCs.

Action From operational mode, issue the **show interfaces at-1/0/6** command.

For brevity, this **show** command output includes only the configuration that is relevant to the at-1/0/6 physical interface. Any other configuration on the system has been replaced with ellipses (...).

```
user@host> show interfaces at-1/0/6
Physical interface: at-1/0/6, Enabled, Physical link is Down
  Interface index: 179, SNMP ifIndex: 598
  Link-level type: ATM-PVC, MTU: 2048, Clocking: Internal, SDH mode, Speed: OC3,
 Loopback: None, Payload scrambler: Enabled
  Device flags   : Present Running Down
  Link flags     : None
  CoS queues     : 8 supported, 8 maximum usable queues
  Schedulers     : 0
  Current address: 00:1f:12:bc:4a:9b
  Last flapped   : 2012-09-19 07:57:59 PDT (07:46:56 ago)
  Input rate     : 0 bps (0 pps)
  Output rate    : 0 bps (0 pps)
  SDH   alarms   : LOL, LOS
  SDH   defects  : LOL, LOS, LOP, BERR-SF, HP-FERF
```

```
    VPI 6
      Flags: Active
      Total down time: 0 sec, Last down: Never
  Traffic statistics:
         Input  packets:                    0
         Output packets:                    0
...
```

Meaning **ATM-PVC** in the Link-level Type field indicates that encapsulation for ATM permanent virtual circuits is being used on ATM physical interface at-1/0/6. The **Active** flag for VPI 6 indicates that the virtual path is up and operational.

Verifying the Encapsulation, VCI, and PPPoE Options Configuration on Logical Unit 2

Purpose Verify that the encapsulation, VCI, and PPPoE settings have been properly configured on logical unit 2 (at-1/0/6.2).

Action From operational mode, issue the **show interfaces at-1/0/6.2** command.

```
user@host> show interfaces at-1/0/6.2
  Logical interface at-1/0/6.2 (Index 345) (SNMP ifIndex 1990)
    Flags: Device-Down Point-To-Point SNMP-Traps 0x4000 Encapsulation:
PPPoE-over-ATM-LLC
    Input packets : 0
    Output packets: 0
    Protocol pppoe
      Dynamic Profile: None,
      Service Name Table: None,
      Max Sessions: 3, Max Sessions VSA Ignore: On,
      Duplicate Protection: On, Short Cycle Protection: mac-address,
      AC Name: ac-pppoeoa
    VCI 6.2
      Flags: Active
      Total down time: 0 sec, Last down: Never
        Input packets : 0
        Output packets: 0
```

Meaning **PPPoE-over-ATM-LLC** in the Encapsulation field indicates that logical interface at-1/0/6.2 is properly configured for PPPoE-over-ATM LLC encapsulation. **Protocol pppoe** indicates that the PPPoE protocol family has been properly configured on the logical interface. The **Active** flag for VCI 6.2 indicates that VCI 2 on VPI 6 is up and operational.

Verifying the Static PPPoE Subscriber Interface Configuration

Purpose Verify that the static PPPoE subscriber interface (pp0.2) is properly configured.

Action From operational mode, issue the **show interfaces pp0** command.

```
user@host> show interfaces pp0
Physical interface: pp0, Enabled, Physical link is Up
  Interface index: 131, SNMP ifIndex: 505
  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

  Logical interface pp0.2 (Index 360) (SNMP ifIndex 1991)
    Flags: Hardware-Down Point-To-Point SNMP-Traps 0x4000 Encapsulation: PPPoE
    PPPoE:
      State: SessionDown, Session ID: None,
      Underlying interface: at-1/0/6.2 (Index 345)
    Input packets : 0
    Output packets: 0
  Keepalive settings: Interval 10 seconds, Up-count 1, Down-count 3
  LCP state: Not-configured
  NCP state: inet: Not-configured, inet6: Not-configured, iso: Not-configured,
mpls: Not-configured
  CHAP state: Closed
  PAP state: Closed
    Protocol inet, MTU: 1492
      Flags: Sendbcast-pkt-to-re, Protocol-Down
      Addresses, Flags: Dest-route-down Is-Preferred Is-Primary
        Destination: 100.0.0/24, Local: 100.0.0.1
```

Meaning **PPPoE** in the Link-level type field indicates that PPPoE encapsulation is in use on the **pp0** physical interface. **PPPoE** in the Encapsulation field indicates that PPPoE encapsulation is also in use on the pp0.2 logical subscriber interface. The Underlying interface field indicates that **at-1/0/6.2** is properly configured as the underlying interface for the static PPPoE subscriber interface. **Protocol inet** indicates that the IPv4 protocol family is properly configured on the pp0.2 logical subscriber interface.

Verifying the PPPoE Underlying Interface Configuration

Purpose Verify that the underlying interface is properly configured for static PPPoE-over-ATM subscriber access.

Action From operational mode, issue the **show pppoe underlying-interfaces at-1/0/6.2 extensive** command.

```
user@host> show pppoe underlying-interfaces at-1/0/6.2 extensive
at-1/0/6.2 Index 345
  State: Static, Dynamic Profile: None,
  Max Sessions: 3, Max Sessions VSA Ignore: On,
  Active Sessions: 0,
  Service Name Table: None,
  Duplicate Protection: On, Short Cycle Protection: mac-address,
  AC Name: ac-pppoeoa,
  PacketType                        Sent         Received
    PADI                               0                0
    PADO                               0                0
    PADR                               0                0
    PADS                               0                0
    PADT                               0                0
    Service name error                 0                0
    AC system error                    0                0
    Generic error                      0                0
    Malformed packets                  0                0
    Unknown packets                    0                0
  Lockout Time (sec):  Min: 120, Max: 240
    Total clients in lockout: 0
    Total clients in lockout grace period: 0
```

Meaning This command indicates that ATM logical interface at-1/0/6.2 is properly configured as the PPPoE underlying interface. **Static** in the State field indicates that at-1/0/0/.2 is statically configured. The remaining fields display information about the PPPoE-specific interface options configured for the PPPoE underlying interface at the **[edit interfaces at-1/0/6 unit 2 family pppoe]** hierarchy level. The Lockout Time fields, which appear in this command only when you display the **extensive** level of output, show the minimum lockout time (120 seconds) and maximum lockout time (240 seconds) configured for the PPPoE underlying interface.

Related Documentation

- ATM for Subscriber Access Overview on page 351
- Configuring ATM for Subscriber Access on page 358
- Example: Configuring a Dynamic PPPoE Subscriber Interface over ATM on page 369
- Example: Configuring a Static Subscriber Interface for IP Access over ATM on page 387
- Example: Configuring a Static Subscriber Interface for IP Access over Ethernet over ATM on page 393
- Example: Configuring a Static PPP Subscriber Interface over ATM on page 399

CHAPTER 37

Configuring Static Subscriber Interfaces over ATM

Example: Configuring a Static Subscriber Interface for IP Access over ATM

This example illustrates a routed IP-over-ATM (IPoA) configuration that creates a subscriber interface for a static IPv4 interface over a static ATM interface on an MX Series router. The router must have Module Port Concentrator/Modular Interface Card (MPC/MIC) interfaces that use an ATM MIC with small form-factor pluggable transceiver (SFP).

Requirements

This example uses the following software and hardware components:

- MX Series 3D Universal Edge Router
- ATM MIC with SFP (Model Number MIC-3D-8OC3-2OC12-ATM) and compatible MPC1 or MPC2

Before you begin:

1. Make sure the MX Series router you are using has an ATM MIC with SFP installed and operational.
   - For information about compatible MPCs for the ATM MIC with SFP, see the *MX Series Interface Module Reference*.

- For information about installing MPCs and MICs in an MX Series router, see the *Hardware Guide* for your MX Series router model.

2. Make sure you understand how to configure and use static ATM interfaces.

   See *ATM Interfaces Overview*.

3. Define the static standard firewall filters (biz-customer-in-filter and biz-customer-out-filter) referenced in the configuration.

   - For information about creating standard firewall filters, see *Guidelines for Configuring Firewall Filters*.
   - For information about applying a firewall filter to an interface, see *Guidelines for Applying Firewall Filters*.

Overview

By using the ATM MIC with SFP and a supported MPC, you can configure the MX Series router to support subscriber access for a statically created IPv4 or IPv6 interface over a static ATM underlying interface. An IPoA configuration enables you to provide access to subscribers on static IPv4 or IPv6 interfaces over an ATM network using ATM Adaptation Layer 5 (AAL5) permanent virtual circuits (PVCs).

NOTE: **IPoA configurations on MX Series routers require static configuration of the IPv4 interface, IPv6 interface, CoS attributes, and firewall fiters. Dynamic configuration is not supported.**

To configure IPoA subscriber access on MX Series routers, you must configure the correct encapsulation type: **atm-snap** for IPoA encapsulation with logical link control (LLC), or **atm-vc-mux** for IPoA encapsulation with virtual circuit (VC) multiplexing. This example configures **atm-vc-mux** as the encapsulation type on the ATM logical interface.

To provision the ATM AAL5 PVCs for access over the ATM network, you must also configure the virtual path identifiers (VPIs) on the ATM physical interface, and one or more virtual circuit identifiers (VCIs) for each VPI.

In IPoA configurations, the subscriber interfaces correspond to the IPv4 or IPv6 addresses that are on the same network as the statically configured ATM underlying interface. In this IPoA example, the IPv4 address 1.0.0.2 represents the subscriber interface. You can configure the destination address with the **set address 1.0.0.2.254/32 destination 1.0.0.2** statement at the **[edit interfaces at-1/0/3 unit 0 family inet]** hierarchy level.

This example includes the following basic steps to statically configure a single IPv4 subscriber interface over an ATM underlying interface:

1. Configure VPI 0 on ATM physical interface at-1/0/3.
2. Configure ATM VC multiplex encapsulation, VCI 0.39 (VCI 39 on VPI 0), and the following IPv4 (**inet**) protocol family characteristics on logical interface at-1/0/3.0 :
   - IP source address validation (**rpf-check**)

- Standard input (biz-customer-in-filter) and output (biz-customer-out-filter) firewall filters
- Interface address 1.0.0.254/32 with destination address 1.0.0.2

3. Configure static access route 200.10.10.0/24 with qualified-next-hop address at-1/0/0.0.

Configuration

To configure a static IPv4 subscriber interface over a static ATM underlying interface, perform these tasks:

- Configuring the ATM Physical Interface on page 389
- Configuring the Static IPv4 Subscriber Interface on Logical Unit 0 on page 390
- Configuring Routing Properties on page 391

CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them in 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.

```
# ATM Physical Interface
set interfaces at-1/0/3 atm-options vpi 0
#
# Logical Unit 0
set interfaces at-1/0/3 unit 0 encapsulation atm-vc-mux
set interfaces at-1/0/3 unit 0 vci 0.39
set interfaces at-1/0/3 unit 0 family inet rpf-check
set interfaces at-1/0/3 unit 0 family inet filter input biz-customer-in-filter
set interfaces at-1/0/3 unit 0 family inet filter output biz-customer-out-filter
set interfaces at-1/0/3 unit 0 family inet address 1.0.0.254/32 destination 1.0.0.2
#
# Routing Properties
set routing-options access route 200.10.10.0/24 qualified-next-hop at-1/0/0.0
```

Configuring the ATM Physical Interface

Step-by-Step Procedure

To configure the ATM physical interface:

1. Specify that you want to configure ATM-specific options on the physical interface.

   ```
   [edit interfaces at-1/0/3]
   user@host# edit atm-options
   ```

2. Configure one or more VPIs on the physical interface.

   ```
   [edit interfaces at-1/0/3 atm-options]
   user@host# set vpi 0
   ```

Results

From the **[edit]** hierarchy level in configuration mode, confirm the results of the ATM physical interface configuration by issuing the **show interfaces at-1/0/3** command. If the output does not display the intended configuration, repeat the instructions in this example to correct it.

```
[edit]
user@host# show interfaces at-1/0/3
atm-options {
  vpi 0;
}
```

If you are done configuring the ATM physical interface, enter **commit** from configuration mode.

Configuring the Static IPv4 Subscriber Interface on Logical Unit 0

Step-by-Step Procedure

To configure the static IPv4 subscriber interface on logical unit 0:

1. Configure ATM VC multiplex encapsulation on the logical interface.

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

2. Configure the VCI for the logical interface.

   ```
   [edit interfaces at-1/0/3 unit 0]
   user@host# set vci 0.39
   ```

3. Configure the IPv4 (**inet**) protocol family, IPv4 address, and remote (destination) address of the connection.

   ```
   [edit interfaces at-1/0/3 unit 0]
   user@host# set family inet address 1.0.0.254/32 destination 1.0.0.2
   ```

4. Specify that you want to configure additional attributes for the IPv4 protocol family.

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

5. Enable IP source address validation, which checks whether traffic is arriving at the router on an expected path.

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

6. Apply the previously defined standard firewall filters to the logical interface.

   ```
   [edit interfaces at-1/0/3 unit 0 family inet]
   user@host# set filter input biz-customer-in-filter
   user@host# set filter output biz-customer-out-filter
   ```

Results

From the **[edit]** hierarchy level in configuration mode, confirm the results of the static subscriber interface configuration on logical unit 0 by issuing the **show interfaces at-1/0/3.0** command. If the output does not display the intended configuration, repeat the instructions in this example to correct it.

```
[edit]
user@host# show interfaces at-1/0/3.0
encapsulation atm-vc-mux;
vci 0.39;
family inet {
  rpf-check;
  filter {
```

```
            input biz-customer-in-filter;
            output biz-customer-out-filter;
        }
        address 1.0.0.254/32 {
            destination 1.0.0.2;
        }
    }
```

If you are done configuring the static subscriber interface on logical unit 0, enter **commit** from configuration mode.

Configuring Routing Properties

Step-by-Step Procedure

To configure static routing properties:

1. Specify that you want to configure protocol-independent routing properties.

   ```
   [edit]
   user@host# edit routing-options
   ```

2. Configure a static access route for routing downstream traffic from the router, and a qualified-next-hop address for routing upstream traffic to the router.

   ```
   [edit routing-options]
   user@host# set access route 200.10.10.0/24 qualified-next-hop at-1/0/0.0
   ```

Results

From the **[edit]** hierarchy level in configuration mode, confirm the results of the static routing properties configuration by issuing the **show routing-options** command. If the output does not display the intended configuration, repeat the instructions in this example to correct it.

```
[edit]
user@host# show routing-options
access {
    route 200.10.10.0/24 {
        qualified-next-hop at-1/0/0.0;
    }
}
```

If you are done configuring the static routing properties, enter **commit** from configuration mode.

Verification

To confirm that the IPoA configuration is working properly, perform the following tasks:

- Verifying the ATM Physical Interface Configuration on page 391
- Verifying the Static Subscriber Interface Configuration on Logical Unit 0 on page 392

Verifying the ATM Physical Interface Configuration

Purpose

Verify that the at-1/0/3 physical interface is properly configured for use with ATM PVCs.

Action

From operational mode, issue the **show interfaces at-1/0/3** command.

For brevity, this **show** command output includes only the configuration that is relevant to the at-1/0/3 physical interface. Any other configuration on the system has been replaced with ellipses (...).

```
user@host> show interfaces at-1/0/3
Physical interface: at-1/0/3, Enabled, Physical link is Down
  Interface index: 168, SNMP ifIndex: 595
  Link-level type: ATM-PVC, MTU: 2048, Clocking: Internal, SONET mode, Speed:
OC3, Loopback: None,
  Payload scrambler: Enabled
  Device flags   : Present Running Down
  Link flags     : None
  CoS queues     : 8 supported, 8 maximum usable queues
  Schedulers     : 0
  Current address: 00:1f:12:bc:4a:98
  Last flapped   : 2012-08-28 07:14:48 PDT (08:28:47 ago)
  Input rate     : 0 bps (0 pps)
  Output rate    : 0 bps (0 pps)
  SONET alarms   : LOL, LOS
  SONET defects  : LOL, LOS, LOP, BERR-SF, RDI-P
    VPI 0
      Flags: Active
      Total down time: 0 sec, Last down: Never
  Traffic statistics:
         Input  packets:                    0
         Output packets:                    0
  ...
```

Meaning **ATM-PVC** in the Link-level Type field indicates that encapsulation for ATM permanent virtual circuits is being used on ATM physical interface at-1/0/3. The **Active** flag for VPI 0 indicates that the virtual path is up and operational.

Verifying the Static Subscriber Interface Configuration on Logical Unit 0

Purpose Verify that the static subscriber interface on logical unit 0 is properly configured for IPv4 access over ATM.

Action From operational mode, issue the **show interfaces at-1/0/3.0** command.

```
user@host> show interfaces at-1/0/3.0
  Logical interface at-1/0/3.0 (Index 341) (SNMP ifIndex 1984)
    Flags: Device-Down Point-To-Point SNMP-Traps 0x4000 Encapsulation: ATM-VCMUX

    Input packets : 0
    Output packets: 0
    Protocol inet, MTU: 2040
      Flags: Sendbcast-pkt-to-re, uRPF
      Addresses, Flags: Dest-route-down Is-Preferred Is-Primary
        Destination: 1.0.0.2, Local: 1.0.0.254
    VCI 0.39
      Flags: Active
      Total down time: 0 sec, Last down: Never
        Input packets : 0
        Output packets: 0
```

Meaning **ATM-VCMUX** in the Encapsulation field indicates that the logical interface at-1/0/3.0 is properly configured for IPoA encapsulation with VC multiplexing. **Protocol inet** indicates that the IPv4 protocol family has been properly configured on the logical interface. The local address 1.0.0.254 is the IPv4 address of the logical interface. The destination address 1.0.0.2, which is in the same network as the local address, is the IPv4 address of the remote side of the connection and represents the static subscriber interface. The **Active** flag for VCI 0.39 indicates that virtual circuit identifier (VCI) 39 on VPI 0 is up and operational.

Related Documentation

- ATM for Subscriber Access Overview on page 351
- Configuring ATM for Subscriber Access on page 358
- Example: Configuring a Dynamic PPPoE Subscriber Interface over ATM on page 369
- Example: Configuring a Static PPPoE Subscriber Interface over ATM on page 378
- Example: Configuring a Static Subscriber Interface for IP Access over Ethernet over ATM on page 393
- Example: Configuring a Static PPP Subscriber Interface over ATM on page 399

Example: Configuring a Static Subscriber Interface for IP Access over Ethernet over ATM

This example illustrates a bridged IP-over-Ethernet-over-ATM (IPoE-over-ATM) configuration that creates a subscriber interface for IPv4 access over a static ATM interface on an MX Series router. The router must have Module Port Concentrator/Modular Interface Card (MPC/MIC) interfaces that use an ATM MIC with small form-factor pluggable transceiver (SFP).

Requirements

This example uses the following software and hardware components:

- MX Series 3D Universal Edge Router
- ATM MIC with SFP (Model Number MIC-3D-8OC3-2OC12-ATM) and compatible MPC1 or MPC2

Before you begin:

1. Make sure the MX Series router you are using has an ATM MIC with SFP installed and operational.
   - For information about compatible MPCs for the ATM MIC with SFP, see the *MX Series Interface Module Reference*.

- For information about installing MPCs and MICs in an MX Series router, see the *Hardware Guide* for your MX Series router model.

2. Make sure you understand how to configure and use static ATM interfaces.

   See *ATM Interfaces Overview*.

3. Define the static standard firewall filters (biz-customer-in-filter and biz-customer-out-filter) referenced in the configuration.

   - For information about creating standard firewall filters, see *Guidelines for Configuring Firewall Filters*.
   - For information about applying a firewall filter to an interface, see *Guidelines for Applying Firewall Filters*.

Overview

By using the ATM MIC with SFP and a supported MPC, you can configure the MX Series router to support subscriber access for a statically created IPv4 or IPv6 interface over a static ATM underlying interface. An IPoE-over-ATM configuration enables you to provide access to subscribers on static IPv4 or IPv6 interfaces over an underlying ATM interface on an ATM network using ATM Adaptation Layer 5 (AAL5) permanent virtual circuits (PVCs).

NOTE: IPoE-over-ATM configurations on MX Series routers require static configuration of the IP interface, ATM interface, CoS attributes, and firewall fiters. Dynamic configuration is not supported.

To configure bridged IPoE-over-ATM subscriber access on MX Series routers, you must configure Ethernet-over-ATM logical link control (LLC) encapsulation on the ATM underlying interface by including the **encapsulation ether-over-atm-llc** statement at the **[edit interfaces *interface-name* unit *logical-unit-number*]** hierarchy level.

To provision the ATM AAL5 PVCs for access over the ATM network, you must also configure the virtual path identifiers (VPIs) on the ATM physical interface, and one or more virtual circuit identifiers (VCIs) for each VPI.

In IPoE-over-ATM configurations, the subscriber interfaces are associated with IPv4 or IPv6 addresses that are mapped to media access control (MAC) addresses. To statically configure Address Resolution Protocol (ARP) table entries that map IP address to MAC addresses, use the **arp** statement at the **[edit interfaces *interface-name* unit *logical-unit-number* family inet address *address*]** hierarchy level. In this example, the IPv4 address 1.0.50.2, configured with the **set arp 1.0.50.2 mac 00:00:01:02:04:ff publish** statement at the **[edit interfaces at-1/0/2 unit 0 family inet address 1.0.50.254/24]** hierarchy level, represents the subscriber interface.

This example includes the following basic steps to statically configure a single IPv4 subscriber interface over an ATM underlying interface:

1. Configure VPI 0 on ATM physical interface at-1/0/2.

2. Configure Ethernet-over-ATM LLC encapsulation, VCI 0.39 (VCI 39 on VPI 0), and the following IPv4 (**inet**) protocol family characteristics on logical interface at-1/0/2.0 :
   - IPv4 subscriber interface address 1.0.50.254/24
   - Static Address Resolution Protocol (ARP) table entries that provide explicit mappings between IP addresses and MAC addresses
   - IP source address validation (**rpf-check**)
   - Standard input (biz-customer-in-filter) and output (biz-customer-out-filter) firewall filters
3. Configure static access route 200.10.10.0/24 with qualified-next-hop address at-1/0/0.0.

Configuration

To configure a static IPv4 subscriber interface over a static ATM underlying interface, perform these tasks:

- Configuring the ATM Physical Interface on page 395
- Configuring the Static IPv4 Subscriber Interface on Logical Unit 0 on page 396
- Configuring Routing Properties on page 397

CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them in 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.

```
# ATM Physical Interface
set interfaces at-1/0/2 atm-options vpi 0
#
# Logical Unit 0
set interfaces at-1/0/2 unit 0 encapsulation ether-over-atm-llc
set interfaces at-1/0/2 unit 0 vci 0.39
set interfaces at-1/0/2 unit 0 family inet rpf-check
set interfaces at-1/0/2 unit 0 family inet filter input biz-customer-in-filter
set interfaces at-1/0/2 unit 0 family inet filter output biz-customer-out-filter
set interfaces at-1/0/2 unit 0 family inet address 1.0.50.254/24 arp 1.0.50.2 mac
   00:00:01:02:04:ff
set interfaces at-1/0/2 unit 0 family inet address 1.0.50.254/24 arp 1.0.50.2 publish
#
# Routing Properties
set routing-options access route 200.10.10.0/24 qualified-next-hop at-1/0/0.0
```

Configuring the ATM Physical Interface

Step-by-Step Procedure

To configure the ATM physical interface:

1. Specify that you want to configure ATM-specific options on the physical interface.

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

2. Configure one or more VPIs on the physical interface.

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

Results

From the **[edit]** hierarchy level in configuration mode, confirm the results of the ATM physical interface configuration by issuing the **show interfaces at-1/0/2** command. If the output does not display the intended configuration, repeat the instructions in this example to correct it.

```
[edit]
user@host# show interfaces at-1/0/2
atm-options {
  vpi 0;
}
```

If you are done configuring the ATM physical interface, enter **commit** from configuration mode.

Configuring the Static IPv4 Subscriber Interface on Logical Unit 0

Step-by-Step Procedure

To configure the static IPv4 subscriber interface on logical unit 0:

1. Configure Ethernet-over-ATM LLC encapsulation on the logical interface.

   ```
   [edit interfaces at-1/0/2 unit 0]
   user@host# set encapsulation ether-over-atm-llc
   ```

2. Configure the VCI for the logical interface.

   ```
   [edit interfaces at-1/0/2 unit 0]
   user@host# set vci 0.39
   ```

3. Configure the IPv4 (**inet**) protocol family and address.

   ```
   [edit interfaces at-1/0/2 unit 0]
   user@host# set family inet address 1.0.50.254/24
   ```

4. Specify that you want to configure static ARP table entries to map between IP addresses and MAC addresses.

   ```
   [edit interfaces at-1/0/2 unit 0 family inet]
   user@host# edit family inet address 1.0.50.254/24
   ```

5. Configure IP address 1.0.50.2, which maps to the MAC address, and MAC address 00:00:01:02:04:ff, which maps to the IP address. Include the **publish** option to specify that the router reply to ARP requests for the specified IP address.

   ```
   [edit interfaces at-1/0/2 unit 0 family inet address 1.0.50.254/24]
   user@host# set arp 1.0.50.2 mac 00:00:01:02:04:ff publish
   user@host# up
   ```

6. Enable IP source address validation, which checks whether traffic is arriving at the router on an expected path.

   ```
   [edit interfaces at-1/0/2 unit 0 family inet]
   user@host# set rpf-check
   ```

7. Apply the previously defined standard firewall filters to the logical interface.


```
[edit interfaces at-1/0/2 unit 0 family inet]
user@host# set filter input biz-customer-in-filter
user@host# set filter output biz-customer-out-filter
```

Results From the **[edit]** hierarchy level in configuration mode, confirm the results of the static subscriber interface configuration on logical unit 0 by issuing the **show interfaces at-1/0/2.0** command. If the output does not display the intended configuration, repeat the instructions in this example to correct it.

```
[edit]
user@host# show interfaces at-1/0/2.0
encapsulation ether-over-atm-llc;
vci 0.39;
family inet {
  rpf-check;
  filter {
    input biz-customer-in-filter;
    output biz-customer-out-filter;
  }
  address 1.0.50.254/24 {
    arp 1.0.50.2 mac 00:00:01:02:04:ff publish;
  }
}
```

If you are done configuring the static subscriber interface on logical unit 0, enter **commit** from configuration mode.

Configuring Routing Properties

Step-by-Step Procedure To configure static routing properties:

1. Specify that you want to configure protocol-independent routing properties.

```
[edit]
user@host# edit routing-options
```

2. Configure a static access route for routing downstream traffic from the router, and a qualified-next-hop address for routing upstream traffic to the router.

```
[edit routing-options]
user@host# set access route 200.10.10.0/24 qualified-next-hop at-1/0/0.0
```

Results From the **[edit]** hierarchy level in configuration mode, confirm the results of the static routing properties configuration by issuing the **show routing-options** command. If the output does not display the intended configuration, repeat the instructions in this example to correct it.

```
[edit]
user@host# show routing-options
access {
  route 200.10.10.0/24 {
    qualified-next-hop at-1/0/0.0;
  }
}
```

If you are done configuring the static routing properties, enter **commit** from configuration mode.

Verification

To confirm that the IPoE-over-ATM configuration is working properly, perform the following tasks:

- Verifying the ATM Physical Interface Configuration on page 398
- Verifying the Static Subscriber Interface Configuration on Logical Unit 0 on page 398

Verifying the ATM Physical Interface Configuration

Purpose Verify that the at-1/0/2 physical interface is properly configured for use with ATM PVCs.

Action From operational mode, issue the **show interfaces at-1/0/2** command.

For brevity, this **show** command output includes only the configuration that is relevant to the at-1/0/2 physical interface. Any other configuration on the system has been replaced with ellipses (...).

```
user@host> show interfaces at-1/0/2
Physical interface: at-1/0/2, Enabled, Physical link is Down
  Interface index: 175, SNMP ifIndex: 594
  Link-level type: ATM-PVC, MTU: 2048, Clocking: Internal, SDH mode, Speed: OC3,
 Loopback: None,
Payload scrambler: Enabled
  Device flags   : Present Running Down
  Link flags     : None
  CoS queues     : 8 supported, 8 maximum usable queues
  Schedulers     : 0
  Current address: 00:1f:12:bc:4a:97
  Last flapped   : 2012-09-06 12:11:39 PDT (05:45:45 ago)
  Input rate     : 0 bps (0 pps)
  Output rate    : 0 bps (0 pps)
  SDH   alarms   : LOL, LOS
  SDH   defects  : LOL, LOS, LOP, BERR-SF, HP-FERF
    VPI 0
      Flags: Active
      Total down time: 0 sec, Last down: Never
  Traffic statistics:
         Input  packets:                    0
         Output packets:                    0
...
```

Meaning **ATM-PVC** in the Link-level Type field indicates that encapsulation for ATM permanent virtual circuits is being used on ATM physical interface at-1/0/2. The **Active** flag for VPI 0 indicates that the virtual path is up and operational.

Verifying the Static Subscriber Interface Configuration on Logical Unit 0

Purpose Verify that the static subscriber interface on logical unit 0 is properly configured for IPoE-over-ATM access.

Action From operational mode, issue the **show interfaces at-1/0/2.0** command.

```
user@host> show interfaces at-1/0/2.0
  Logical interface at-1/0/2.0 (Index 336) (SNMP ifIndex 1983)
    Flags: Device-Down Point-To-Multipoint SNMP-Traps 0x4000 Encapsulation:
Ether-over-ATM-LLC
    Input packets : 0
    Output packets: 0
    Protocol inet, MTU: 2016
      Flags: Sendbcast-pkt-to-re, uRPF
      Addresses, Flags: Dest-route-down Is-Preferred Is-Primary
        Destination: 1.0.50/24, Local: 1.0.50.254, Broadcast: 1.0.50.255
    VCI 0.39
      Flags: Active, Multicast
      Total down time: 0 sec, Last down: Never
        Input packets : 0
        Output packets: 0
```

Meaning **Ether-over-ATM-LLC** in the Encapsulation field indicates that logical interface at-1/0/2.0 is properly configured for Ethernet-over-ATM encapsulation with LLC. **Protocol inet** indicates that the IPv4 protocol family has been properly configured on the logical interface. The destination address 1.0.50/24 identifies the network in which the subscriber interface (1.0.50.2) resides. The **Active** flag for VCI 0.39 indicates that virtual circuit identifier (VCI) 39 on VPI 0 is up and operational.

Related Documentation

- ATM for Subscriber Access Overview on page 351
- Configuring ATM for Subscriber Access on page 358
- Example: Configuring a Dynamic PPPoE Subscriber Interface over ATM on page 369
- Example: Configuring a Static PPPoE Subscriber Interface over ATM on page 378
- Example: Configuring a Static Subscriber Interface for IP Access over ATM on page 387
- Example: Configuring a Static PPP Subscriber Interface over ATM on page 399

Example: Configuring a Static PPP Subscriber Interface over ATM

This example illustrates a PPP-over-ATM (PPPoA) configuration that creates three static PPP logical subscriber interfaces over a static ATM underlying interface on an MX Series router. The router must have Module Port Concentrator/Modular Interface Card (MPC/MIC) interfaces that use an ATM MIC with small form-factor pluggable transceiver (SFP).

Requirements

This example uses the following software and hardware components:

- MX Series 3D Universal Edge Router
- ATM MIC with SFP (Model Number MIC-3D-8OC3-2OC12-ATM) and compatible MPC1 or MPC2

Before you begin:

1. Make sure the MX Series router you are using has an ATM MIC with SFP installed and operational.
   - For information about compatible MPCs for the ATM MIC with SFP, see the *MX Series Interface Module Reference*.
   - For information about installing MPCs and MICs in an MX Series router, see the *Hardware Guide* for your MX Series router model.
2. Make sure you understand how to configure and use static ATM interfaces.

   See *ATM Interfaces Overview*.
3. Create the dynamic profile (pppoa-cos-profile) and access profile (pe-B-ppp-clients) referenced in the configuration.
   - For information about creating a basic dynamic profile, see *Configuring a Basic Dynamic Profile*.
   - For information about creating a dynamic profile for class of service (CoS) attributes, see *Configuring Traffic Scheduling and Shaping for Subscriber Access*.
   - For information about creating an access profile for PPP Challenge Handshake Authentication Protocol (CHAP) authentication, see *Configuring the PPP Challenge Handshake Authentication Protocol*.

Overview

By using the ATM MIC with SFP and a supported MPC, you can configure an MX Series router to support PPP subscriber access over an ATM network. PPPoA configurations on MX Series routers consist of one or more statically created PPP logical subscriber interfaces over a static ATM underlying interface.

Optionally, you can use dynamic profiles to dynamically or statically apply subscriber services services, such as CoS and firewall filters, to the static PPP logical interface. Configuring CoS and firewall filters in this manner enables you to efficiently and economically provide these services to PPP subscribers accessing the router over an ATM network using ATM Adaptation Layer 5 (AAL5) permanent virtual connections (PVCs). This example uses a previously configured dynamic profile named pppoa-cos-profile to apply traffic scheduling and shaping parameters to logical interface at-1/0/1.2.

To configure PPPoA subscriber access on MX Series routers, you must configure the correct encapsulation type: **atm-ppp-llc** for PPPoA encapsulation with logical link control (LLC), or **atm-ppp-vc-mux** for PPPoA encapsulation with virtual circuit (VC) multiplexing. This example configures **atm-ppp-llc** as the encapsulation type on logical interface at-1/0/1.0, and **atm-ppp-vc-mux** as the encapsulation type on logical interfaces at-1/0/1.1 and at-1/0/1.2.

To provision the ATM AAL5 PVCs for access over the ATM network, you must also configure the virtual path identifiers (VPIs) on the ATM physical interface, and one or more virtual circuit identifiers (VCIs) for each VPI.

In PPPoA configurations, each statically configured logical interface (for example, at-1/0/1.0) corresponds to a PPP logical subscriber interface. This example configures three PPP logical subscriber interfaces over an ATM interface, as follows:

- The ATM physical interface (at-1/0/1) is statically configured with VPI 0 and VPI 2.
- Logical interface at-1/0/1.0 (logical unit 0) is configured with PPP-over AAL5 LLC encapsulation, VCI 0.120 (VCI 120 on VPI 0), PPP-specific options, and the IPv4 protocol family and address.
- Logical interface at-1/0/1.1 (logical unit 1) is configured with PPP-over-AAL5 VC multiplexing encapsulation, VCI 2.120 (VCI 120 on VPI 2), PPP-specific options, and the IPv4 protocol family and address.
- Logical interface at-1/0/1.2 (logical unit 2) is configured with PPP-over-AAL5 VC multiplexing encapsulation, VCI 2.121 (VCI 121 on VPI 2), PPP-specific options, and the IPv4 protocol family and address. The PPP-specific options include applying a dynamic profile named pppoa-cos-profile to the static PPP interface. The pppoa-cos-profile dynamic profile applies traffic scheduling and shaping parameters to the PPP logical subscriber interface.

Configuration

To configure static PPP logical subscriber interfaces over an ATM interface, perform these tasks:

- Configuring the ATM Physical Interface on page 402
- Configuring the Static PPP Subscriber Interface on Logical Unit 0 on page 403
- Configuring the Static PPP Subscriber Interface on Logical Unit 1 on page 404
- Configuring the Static PPP Subscriber Interface on Logical Unit 2 on page 405

CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them in 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.

```
# ATM Physical Interface
set interfaces at-1/0/1 atm-options vpi 0
set interfaces at-1/0/1 atm-options vpi 2
#
# Logical Unit 0
```

```
set interfaces at-1/0/1 unit 0 encapsulation atm-ppp-llc
set interfaces at-1/0/1 unit 0 vci 0.120
set interfaces at-1/0/1 unit 0 ppp-options chap access-profile pe-B-ppp-clients
set interfaces at-1/0/1 unit 0 ppp-options chap local-name pe-A-at-1/0/1
set interfaces at-1/0/1 unit 0 keepalives interval 5
set interfaces at-1/0/1 unit 0 keepalives up-count 6
set interfaces at-1/0/1 unit 0 keepalives down-count 4
set interfaces at-1/0/1 unit 0 family inet address 192.122.13.13/30
#
# Logical Unit 1
set interfaces at-1/0/1 unit 1 encapsulation atm-ppp-vc-mux
set interfaces at-1/0/1 unit 1 vci 2.120
set interfaces at-1/0/1 unit 1 keepalives interval 6
set interfaces at-1/0/1 unit 1 keepalives up-count 6
set interfaces at-1/0/1 unit 1 keepalives down-count 4
set interfaces at-1/0/1 unit 1 family inet address 192.122.14.13/30
#
# Logical Unit 2
set interfaces at-1/0/1 unit 2 encapsulation atm-ppp-vc-mux
set interfaces at-1/0/1 unit 2 vci 2.121
set interfaces at-1/0/1 unit 2 ppp-options chap access-profile pe-A-ppp-clients
set interfaces at-1/0/1 unit 2 ppp-options chap local-name pe-A-at-1/0/1
set interfaces at-1/0/1 unit 2 ppp-options chap passive
set interfaces at-1/0/1 unit 2 ppp-options dynamic-profile pppoa-cos-profile
set interfaces at-1/0/1 unit 2 keepalives interval 5
set interfaces at-1/0/1 unit 2 keepalives up-count 6
set interfaces at-1/0/1 unit 2 keepalives down-count 4
set interfaces at-1/0/1 unit 2 family inet address 192.122.15.13/30
```

Configuring the ATM Physical Interface

Step-by-Step Procedure

To configure the ATM physical interface:

1. Specify that you want to configure ATM-specific options on the physical interface.

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

2. Configure one or more VPIs on the physical interface.

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

Results

From the **[edit]** hierarchy level in configuration mode, confirm the results of the ATM physical interface configuration by issuing the **show interfaces at-1/0/1** command. If the output does not display the intended configuration, repeat the instructions in this example to correct it.

```
[edit]
user@host# show interfaces at-1/0/1
atm-options {
  vpi 0;
  vpi 2;
}
```

If you are done configuring the ATM physical interface, enter **commit** from configuration mode.

Configuring the Static PPP Subscriber Interface on Logical Unit 0

Step-by-Step Procedure

To configure the static PPP subscriber interface on logical unit 0:

1. Configure PPP-over AAL5 LLC encapsulation on the logical interface.

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

2. Configure the VCI for the logical interface.

   ```
   [edit interfaces at-1/0/1 unit 0]
   user@host# set vci 0.120
   ```

3. Specify that you want to configure options for PPP CHAP on the logical interface.

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

4. Assign the previously configured pe-B-ppp-clients access profile to the PPP logical subscriber interface.

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

5. Configure the local name used by the interface in CHAP challenge and response packets.

   ```
   [edit interfaces at-1/0/1 unit 0 ppp-options chap]
   user@host# set local-name "pe-A-at-1/0/1"
   user@host# up 2
   ```

6. Configure the transmission of keepalive messages on the logical interface.

   ```
   [edit interfaces at-1/0/1 unit 0]
   user@host# set keepalives interval 5
   user@host# set keepalives up-count 6
   user@host# set keepalives down-count 4
   ```

7. Configure the IPv4 (**inet**) protocol family and IP address.

   ```
   [edit interfaces at-1/0/1 unit 0]
   user@host# set family inet address 192.122.13.13/30
   ```

Results

From the **[edit]** hierarchy level in configuration mode, confirm the results of the static PPP subscriber interface configuration on logical unit 0 by issuing the **show interfaces at-1/0/1.0** command. If the output does not display the intended configuration, repeat the instructions in this example to correct it.

```
[edit]
user@host# show interfaces at-1/0/1.0
encapsulation atm-ppp-llc;
vci 0.120;
ppp-options {
  chap {
    access-profile pe-B-ppp-clients;
```

```
        local-name pe-A-at-1/0/1;
      }
    }
    keepalives interval 5 up-count 6 down-count 4;
    family inet {
      address 192.122.13.13/30;
    }
```

If you are done configuring the PPP logical subscriber interface on logical unit 0, enter **commit** from configuration mode.

Configuring the Static PPP Subscriber Interface on Logical Unit 1

Step-by-Step Procedure

To configure the static PPP subscriber interface on logical unit 1:

1. Configure PPP-over-AAL5 VC multiplexing encapsulation on the logical interface.

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

2. Configure the VCI for the logical interface.

   ```
   [edit interfaces at-1/0/1 unit 1]
   user@host# set vci 2.120
   ```

3. Configure the transmission of keepalive messages on the logical interface.

   ```
   [edit interfaces at-1/0/1 unit 1]
   user@host# set keepalives interval 6
   user@host# set keepalives up-count 6
   user@host# set keepalives down-count 4
   ```

4. Configure the IPv4 (**inet**) protocol family and IP address.

   ```
   [edit interfaces at-1/0/1 unit 1]
   user@host# set family inet address 192.122.14.13/30
   ```

Results

From the **[edit]** hierarchy level in configuration mode, confirm the results of the static PPP subscriber interface configuration on logical unit 1 by issuing the **show interfaces at-1/0/1.1** command. If the output does not display the intended configuration, repeat the instructions in this example to correct it.

```
[edit]
user@host# show interfaces at-1/0/1.1
encapsulation atm-ppp-vc-mux;
vci 2.120;
keepalives interval 6 up-count 6 down-count 4;
family inet {
  address 192.122.14.13/30;
}
```

If you are done configuring the PPP logical subscriber interface on logical unit 1, enter **commit** from configuration mode.

Configuring the Static PPP Subscriber Interface on Logical Unit 2

Step-by-Step Procedure

To configure the static PPP subscriber interface on logical unit 2:

1. Configure PPP-over-AAL5 VC multiplex encapsulation on the logical interface.

   ```
   [edit interfaces at-1/0/1 unit 2]
   user@host# set encapsulation atm-ppc-vc-mux
   ```

2. Configure the VCI for the logical interface.

   ```
   [edit interfaces at-1/0/1 unit 2]
   user@host# set vci 2.121
   ```

3. Specify that you want to configure options for PPP CHAP on the logical interface.

   ```
   [edit interfaces at-1/0/1 unit 2]
   user@host# edit ppp-options chap
   ```

4. Assign the previously configured pe-A-ppp-clients access profile to the PPP logical subscriber interface.

   ```
   [edit interfaces at-1/0/1 unit 2 ppp-options chap]
   user@host# set access-profile pe-A-ppp-clients
   ```

5. Configure the local name used by the interface in CHAP challenge and response packets.

   ```
   [edit interfaces at-1/0/1 unit 2 ppp-options chap]
   user@host# set local-name "pe-A-at-1/0/1"
   ```

6. Configure passive mode for CHAP authentication.

   ```
   [edit interfaces at-1/0/1 unit 2 ppp-options chap]
   user@host# set passive
   user@host# up
   ```

7. Apply the previously configured pppoa-cos-profile dynamic profile to the PPP logical subscriber interface.

   ```
   [edit interfaces at-1/0/1 unit 2 ppp-options]
   user@host# set dynamic-profile pppoa-cos-profile
   user@host# up
   ```

8. Configure the transmission of keepalive messages on the logical interface.

   ```
   [edit interfaces at-1/0/1 unit 2]
   user@host# set keepalives interval 5
   user@host# set keepalives up-count 6
   user@host# set keepalives down-count 4
   ```

9. Configure the IPv4 (**inet**) protocol family and IP address.

   ```
   [edit interfaces at-1/0/1 unit 2]
   user@host# set family inet address 192.122.15.13/30
   ```

Results

From the **[edit]** hierarchy level in configuration mode, confirm the results of the static PPP subscriber interface configuration on logical unit 2 by issuing the **show interfaces at-1/0/1.2** command. If the output does not display the intended configuration, repeat the instructions in this example to correct it.

```
[edit]
user@host# show interfaces at-1/0/1.2
encapsulation atm-ppp-vc-mux;
vci 2.121;
ppp-options {
  chap {
    access-profile pe-A-ppp-clients;
    local-name pe-A-at-1/0/1;
    passive;
  }
  dynamic-profile pppoa-cos-profile;
}
keepalives interval 5 up-count 6 down-count 4;
family inet {
  address 192.122.15.13/30;
}
```

If you are done configuring the PPP logical subscriber interface on logical unit 2, enter **commit** from configuration mode.

Verification

To confirm that the PPPoA configuration is working properly, perform the following tasks:

- Verifying the ATM Physical Interface Configuration on page 406
- Verifying the Static PPPoA Configuration on Logical Unit 0 on page 407
- Verifying the Static PPPoA Configuration on Logical Unit 1 on page 408
- Verifying the Static PPPoA Configuration on Logical Unit 2 on page 408

Verifying the ATM Physical Interface Configuration

Purpose Verify that the at-1/0/1 physical interface is properly configured for use with ATM PVCs.

Action From operational mode, issue the **show interfaces at-1/0/1** command.

For brevity, this **show** command output includes only the configuration that is relevant to the at-1/0/1 physical interface. Any other configuration on the system has been replaced with ellipses (...).

```
user@host> show interfaces at-1/0/1
Physical interface: at-1/0/1, Enabled, Physical link is Down
  Interface index: 166, SNMP ifIndex: 593
  Link-level type: ATM-PVC, MTU: 2048, Clocking: Internal, SONET mode, Speed:
OC3, Loopback: None, Payload scrambler: Enabled
  Device flags   : Present Running Down
  Link flags     : None
  CoS queues     : 8 supported, 8 maximum usable queues
  Schedulers     : 0
  Current address: 00:1f:12:bc:4a:96
  Last flapped   : 2012-06-29 15:35:29 PDT (2d 16:24 ago)
  Input rate     : 0 bps (0 pps)
  Output rate    : 0 bps (0 pps)
  SONET alarms   : LOL, LOS
  SONET defects  : LOL, LOS, LOP, BERR-SF, RDI-P
    VPI 0
```



```
      Flags: Active
      Total down time: 0 sec, Last down: Never
  Traffic statistics:
         Input  packets:                    0
         Output packets:                    0
    VPI 2
      Flags: Active
      Total down time: 0 sec, Last down: Never
  Traffic statistics:
         Input  packets:                    0
         Output packets:                    0

  ...
```

Meaning **ATM-PVC** in the Link-level Type field indicates that encapsulation for ATM permanent virtual circuits is being used on ATM physical interface at-1/0/1. The **Active** flag for VPI 0 and VPI 2 indicates that these virtual paths are up and operational.

Verifying the Static PPPoA Configuration on Logical Unit 0

Purpose Verify that the static PPP subscriber interface is properly configured on logical unit 0 (at-1/0/1.0).

Action From operational mode, issue the **show interfaces at-1/0/1.0** command.

```
user@host> show interfaces at-1/0/1.0
Logical interface at-1/0/1.0 (Index 337) (SNMP ifIndex 1979)
    Flags: Device-Down Point-To-Point Inverse-ARP SNMP-Traps 0x4000 Encapsulation:
 ATM-PPP-LLC
     Input packets : 0
     Output packets: 0
  Keepalive settings: Interval 5 seconds, Up-count 6, Down-count 4
    LCP state: Down
  NCP state: inet: Not-configured, inet6: Not-configured, iso: Not-configured,
mpls: Not-configured
  CHAP state: Closed
  PAP state: Closed
    Protocol inet, MTU: 2034
      Flags: Sendbcast-pkt-to-re, Protocol-Down
      Addresses, Flags: Dest-route-down Is-Preferred Is-Primary
        Destination: 192.122.13.12/30, Local: 192.122.13.13, Broadcast:
192.122.13.15
    VCI 0.120
      Flags: Active, Inverse-ARP
      Total down time: 0 sec, Last down: Never
      ARP statistics
       Received: 0, Sent: 0, Denied: 0, Operation not supported: 0,
       Bad packet length: 0, Bad protocol: 0, Bad protocol length: 0,
       Bad hardware length: 0, Dropped: 0
       Last received: Never, Last sent: Never
        Input packets : 0
        Output packets: 0
```

Meaning **ATM-PPP-LLC** in the Encapsulation field indicates that logical interface at-1/0/1.0 is properly configured for PPP-over-AAL5 logical link control (LLC) encapsulation. **Protocol inet** indicates that the IPv4 protocol family has been properly configured on the logical interface. The **Active** flag for VCI 0.120 indicates that virtual circuit identifier (VCI) 120 on VPI 0 is up and operational.

Verifying the Static PPPoA Configuration on Logical Unit 1

Purpose Verify that the static PPP subscriber interface is properly configured on logical unit 1 (at-1/0/1.1).

Action From operational mode, issue the **show interfaces at-1/0/1.1** command.

```
user@host> show interfaces at-1/0/1.1

  Logical interface at-1/0/1.1 (Index 338) (SNMP ifIndex 1980)
    Flags: Device-Down Point-To-Point SNMP-Traps 0x4000 Encapsulation:
ATM-PPP-VCMUX
    Input packets : 0
    Output packets: 0
  Keepalive settings: Interval 6 seconds, Up-count 6, Down-count 4
   LCP state: Down
  NCP state: inet: Not-configured, inet6: Not-configured, iso: Not-configured,
mpls: Not-configured
  CHAP state: Closed
  PAP state: Closed
    Protocol inet, MTU: 2038
      Flags: Sendbcast-pkt-to-re, Protocol-Down
      Addresses, Flags: Dest-route-down Is-Preferred Is-Primary
        Destination: 192.122.14.12/30, Local: 192.122.14.13, Broadcast:
192.122.14.15
    VCI 2.120
      Flags: Active, Inverse-ARP
      Total down time: 0 sec, Last down: Never
      ARP statistics
       Received: 0, Sent: 0, Denied: 0, Operation not supported: 0,
       Bad packet length: 0, Bad protocol: 0, Bad protocol length: 0,
       Bad hardware length: 0, Dropped: 0
       Last received: Never, Last sent: Never
        Input packets : 0
        Output packets: 0
```

Meaning **ATM-PPP-VCMUX** in the Encapsulation field indicates that the logical interface at-1/0/1.1 is properly configured for PPP-over-AAL5 VC multiplexing encapsulation. **Protocol inet** indicates that the IPv4 protocol family has been properly configured on the logical interface. The **Active** flag for VCI 2.120 indicates that virtual circuit identifier (VCI) 120 on VPI 2 is up and operational.

Verifying the Static PPPoA Configuration on Logical Unit 2

Purpose Verify that the static PPP subscriber interface is properly configured on logical unit 2 (at-1/0/1.2).

Action From operational mode, issue the **show interfaces at-1/0/1.2** command.

```
user@host> show interfaces at-1/0/1.2
Logical interface at-1/0/1.2 (Index 339) (SNMP ifIndex 1981)
    Flags: Device-Down Point-To-Point SNMP-Traps 0x4000 Encapsulation:
ATM-PPP-VCMUX
    Input packets : 0
    Output packets: 0
  Keepalive settings: Interval 5 seconds, Up-count 6, Down-count 4
   LCP state: Down
  NCP state: inet: Not-configured, inet6: Not-configured, iso: Not-configured,
mpls: Not-configured
  CHAP state: Closed
  PAP state: Closed
    Protocol inet, MTU: 2038
      Flags: Sendbcast-pkt-to-re, Protocol-Down
      Addresses, Flags: Dest-route-down Is-Preferred Is-Primary
        Destination: 192.122.15.12/30, Local: 192.122.15.13, Broadcast:
192.122.15.15
    VCI 2.121
      Flags: Active
      Total down time: 0 sec, Last down: Never
        Input packets : 0
        Output packets: 0
```

Meaning **ATM-PPP-VCMUX** in the Encapsulation field indicates that the logical interface at-1/0/1.2 is properly configured for PPP-over-AAL5 VC multiplexing encapsulation. **Protocol inet** indicates that the IPv4 protocol family has been properly configured on the logical interface. The **Active** flag for VCI 2.121 indicates that virtual circuit identifier 121 on VPI 2 is up and operational.

Related Documentation

- ATM for Subscriber Access Overview on page 351
- Configuring ATM for Subscriber Access on page 358
- Example: Configuring a Dynamic PPPoE Subscriber Interface over ATM on page 369
- Example: Configuring a Static PPPoE Subscriber Interface over ATM on page 378
- Example: Configuring a Static Subscriber Interface for IP Access over ATM on page 387
- Example: Configuring a Static Subscriber Interface for IP Access over Ethernet over ATM on page 393

PART 7

Troubleshooting

CHAPTER 38

Contacting Juniper Networks Technical Support

Collecting Subscriber Access Logs Before Contacting Juniper Networks Technical Support

Problem

Description: When you experience a subscriber access problem in your network, we recommend that you collect certain logs before you contact Juniper Networks Technical Support. This topic shows you the most useful logs for a variety of network implementations. In addition to the relevant log information, you must also collect standard troubleshooting information and send it to Juniper Networks Technical Support in your request for assistance.

Solution

To collect standard troubleshooting information:

- Redirect the command output to a file.

  user@host> **request support information | save rsi-1**

To configure logging to assist Juniper Networks Technical Support:

1. Review the following blocks of statements to determine which apply to your configuration.

```
[edit]
set system syslog archive size 100m files 25
set system auto-configuration traceoptions file filename
set system auto-configuration traceoptions file filename size 100m files 25
set protocols ppp-service traceoptions file filename size 100m files 25
set protocols ppp-service traceoptions level all
set protocols ppp-service traceoptions flag all
set protocols ppp traceoptions file filename size 100m files 25
set protocols ppp traceoptions level all
set protocols ppp traceoptions flag all
set protocols ppp monitor-session all
set interfaces pp0 traceoptions flag all
set demux traceoptions file filename size 100m files 25
set demux traceoptions level all
set demux traceoptions flag all
set system processes dhcp-service traceoptions file filename
set system processes dhcp-service traceoptions file size 100m
set system processes dhcp-service traceoptions file files 25
set system processes dhcp-service traceoptions flag all
set class-of-service traceoptions file filename
set class-of-service traceoptions file size 100m
set class-of-service traceoptions flag all
set class-of-service traceoptions file files 25
set routing-options traceoptions file filename
set routing-options traceoptions file size 100m
set routing-options traceoptions flag all
set routing-options traceoptions file files 25
set interfaces traceoptions file filename
set interfaces traceoptions file size 100m
set interfaces traceoptions flag all
set interfaces traceoptions file files 25
set system processes general-authentication-service traceoptions file filename
set system processes general-authentication-service traceoptions file size 100m
set system processes general-authentication-service traceoptions flag all
set system processes general-authentication-service traceoptions file files 25
```

2. Copy the relevant statements into a text file and modify the log filenames as you want.

3. Copy the statements from the text file and paste them into the CLI on your router to configure logging.

4. Commit the logging configuration to begin collecting information.

NOTE: The maximum file size for DHCP local server and DHCP relay log files is 1 GB. The maximum number of log files for DHCP local server and DHCP relay is 1000.

BEST PRACTICE: Enable these logs only to collect information when troubleshooting specific problems. Enabling these logs during normal operations can result in reduced system performance.

Related Documentation

- *Compressing Troubleshooting Logs from /var/logs to Send to Juniper Networks Technical Support*

PART 8

Configuration Statements and Operational Commands

CHAPTER 39

Configuration Statements

[edit dynamic-profiles] Hierarchy Level

```
dynamic-profiles {
  profile-name {
    class-of-service {
      interfaces {
        interface-name {
          unit logical-unit-number {
            classifiers {
              type (classifier-name | default);
            }
            output-traffic-control-profile (profile-name |
              $junos-cos-traffic-control-profile);
            rewrite-rules {
              dscp (rewrite-name | default);
              dscp-ipv6 (rewrite-name | default);
              ieee-802.1 (rewrite-name | default) vlan-tag (outer | outer-and-inner);
              inet-precedence (rewrite-name | default);
              }
            }
          }
        }
      }
      scheduler-maps {
        map-name {
          forwarding-class class-name scheduler scheduler-name;
        }
      }
      schedulers {
        (scheduler-name) {
          buffer-size (percent percentage | remainder | temporal microseconds |
            $junos-cos-scheduler-bs);
```

```
                drop-profile-map loss-priority (any | low | medium-low | medium-high | high)
                    protocol (any | non-tcp | tcp) drop-profile (profile-name | predefined-variable);
                excess-priority (low | high | $junos-cos-scheduler-excess-priority);
                excess-rate (percent percentage  | percent $junos-cos-scheduler-excess-rate);
                overhead-accounting (shaping-mode) <bytes (byte-value>;
                priority (priority-level | $junos-cos-scheduler-priority);
                shaping-rate (rate | predefined-variable);
                transmit-rate (rate | percent percentage | remainder | percent percentage
                    $junos-cos-scheduler-tx) <exact | rate-limit>;
            }
        }
        traffic-control-profiles profile-name {
            delay-buffer-rate (percent percentage | rate);
            excess-rate (percent percentage | proportion value | percent
                $junos-cos-excess-rate);
            guaranteed-rate (percent percentage | rate);
            overhead-accounting (shaping-mode) <bytes (byte-value>;
            scheduler-map map-name;
            shaping-rate (percent percentage | rate | predefined-variable);
        }
    }
    firewall {
        family family {
            fast-update-filter filter-name {
                interface-specific;
            match-order [match-order];
                term term-name {
                    from {
                        match-conditions;
                    }
                    then {
                        action;
                        action-modifiers;
                    }
                only-at-create;
            filter filter-name {
            interface-specific;
                term term-name {
                    from {
                        match-conditions;
                    }
                    then {
                        action;
                        action-modifiers;
                    }
                }
        policer policer-name {
            filter-specific;
            if-exceeding {
                (bandwidth-limit bps | bandwidth-percent percentage);
                burst-size-limit bytes;
            }
            logical-bandwidth-policer;
            logical-interface-policer;
            physical-interface-policer;
            then {
```

```
            policer-action;
          }
        }
        hierarchical-policer policer-name {
          aggregate {
            if-exceeding {
              bandwidth-limit-limit bps;
              burst-size-limit bytes;
            }
            then {
              policer-action;
            }
          }
          premium {
            if-exceeding {
              bandwidth-limit bps;
              burst-size-limit bytes;
            }
            then {
              policer-action;
            }
          }
        }
        three-color-policer policer-name {
          action {
            loss-priority high then discard;
          }
          logical-interface-policer;
          single-rate {
            (color-aware | color-blind);
            committed-burst-size bytes;
            committed-information-rate bps;
            excess-burst-size bytes;
          }
          two-rate {
            (color-aware | color-blind);
            committed-burst-size bytes;
            committed-information-rate bps;
            peak-burst-size bytes;
            peak-information-rate bps;
            }
          }
        }
      }
      policy-options {
        prefix-listname {
          ip-addresses;
          dynamic-db;
        }
      }
      interfaces {
        interface-name {
          unit logical-unit-number {
            family family {
              access-concentrator name;
              address address;
```

```
            direct-connect;
            duplicate-protection;
            dynamic-profile profile-name;
            filter {
              adf {
                counter;
                input-precedence precedence;
                not-mandatory;
                output-precedence precedence;
                rule rule-value;
              }
              input filter-name {
                precedence precedence;
                shared-name filter-shared-name;
              }
              output filter-name {
                precedence precedence;
                shared-name filter-shared-name;
              }
            }
            max-sessions number;
            max-sessions-vsa-ignore;
            rpf-check {
              fail-filter filter-name;
              mode loose;
            }
            service {
              input {
                service-set service-set-name {
                  service-filter filter-name;
                }
                post-service-filter filter-name;
              }
              output {
                service-set service-set-name {
                  service-filter filter-name;
                }
              }
            }
            service-name-table table-name;
            short-cycle-protection <lockout-time-min minimum-seconds lockout-time-max
              maximum-seconds>;
            unnumbered-address interface-name <preferred-source-address address>;
          }
          ppp-options {
            chap;
            pap;
          }
          vlan-id number;
        }
        vlan-tagging;
      }
      interface-set interface-set-name {
        interface interface-name {
          unit logical-unit-number;
        }
```



```
}
demux0 {
  unit logical-unit-number {
    demux-options {
      underlying-interface interface-name
    }
    demux-source {
      source-prefix;
    }
    family family {
      access-concentrator name;
      address address;
      direct-connect;
      duplicate-protection;
      dynamic-profile profile-name;
      filter {
        input filter-name;
        output filter-name;
      }
      mac-validate (loose | strict):
      max-sessions number;
      max-sessions-vsa-ignore;
      service-name-table table-name;
      short-cycle-protection <lockout-time-min minimum-seconds lockout-time-max
        maximum-seconds>;
      unnumbered-address interface-name <preferred-source-address address>;
    }
  }
}
pp0 {
  unit logical-unit-number {
    keepalives interval seconds;
    no-keepalives;
    pppoe-options {
      underlying-interface interface-name;
      server;
    }
    ppp-options {
      authentication [ authentication-protocols ];
      chap {
        challenge-length minimum minimum-length maximum maximum-length;
      }
      pap;
    }
    family inet {
      unnumbered-address interface-name;
      address address;
      service {
        input {
          service-set service-set-name {
            service-filter filter-name;
          }
          post-service-filter filter-name;
        }
        output {
          service-set service-set-name {
```



```
                    service-filter filter-name;
                  }
                }
              }
              filter {
                input filter-name {
                  precedence precedence;
                }
                output filter-name {
                  precedence precedence;
                }
              }
            }
          }
        }
      }
      protocols {
        igmp {
          interface interface-name {
            accounting;
            disable;
            group-policy;
            immediate-leave
            no-accounting;
            promiscuous-mode;
            ssm-map ssm-map-name;
            static {
              group group {
                source source;
              }
            }
            version version;
          }
          mld {
            interface interface-name {
              disable;
              (accounting | no-accounting);
              group-policy;
              immediate-leave;
              oif-map;
              passive;
              ssm-map ssm-map-name;
              static {
                group multicast-group-address {
                  exclude;
                  group-count number;
                  group-increment increment;
                  source ip-address {
                    source-count number;
                    source-increment increment;
                  }
                }
              }
              version version;
            }
          }
```



```
            router-advertisement {
              interface interface-name {
                current-hop-limit number;
                default-lifetime seconds;
                (managed-configuration | no-managed-configuration);
                max-advertisement-interval seconds;
                min-advertisement-interval seconds;
                (other-stateful-configuration | no-other-stateful-configuration);
                prefix prefix {
                  (autonomous | no-autonomous);
                  (on-link | no-on-link);
                  preferred-lifetime seconds;
                  valid-lifetime seconds;
                }
                reachable-time milliseconds;
                retransmit-timer milliseconds;
              }
            }
          }
        }
      }
      routing-instances routing-instance-name {
        interface interface-name;
        routing-options {
          access {
            route prefix {
              next-hop next-hop;
              metric route-cost;
              preference route-distance;
              tag route-tag;
            }
          }
          access-internal {
            route subscriber-ip-address {
              qualified-next-hop underlying-interface {
                mac-address address;
              }
            }
          }
          multicast {
            interface interface-name {
              no-qos-adjust;
            }
          }
        }
        rib routing-table-name {
          access {
            route prefix {
              next-hop next-hop;
              metric route-cost;
              preference route-distance;
              tag route-tag;
            }
          }
          access-internal {
            route subscriber-ip-address {
```

```
                qualified-next-hop underlying-interface {
                    mac-address address;
                }
            }
        }
    }
}
routing-options {
    access {
        route prefix {
            next-hop next-hop;
            metric route-cost;
            preference route-distance;
            tag route-tag;
        }
    }
    access-internal {
        route subscriber-ip-address {
            qualified-next-hop underlying-interface {
                mac-address address;
            }
        }
    }
    multicast {
        interface interface-name {
            no-qos-adjust;
        }
    }
}
variables {
    variable-name {
        default-value default-value;
        equals expression;
        mandatory;
        uid;
        uid-reference;
    }
}
}
```

Related Documentation

- *Dynamic Profiles Overview*
- *CoS for Subscriber Access Overview*
- *Configuring a Basic Dynamic Profile*
- *Configuring Static Hierarchical Scheduling and Queuing in a Dynamic Profile for Subscriber Access*
- *Two-Color Policer Configuration Overview*
- *Three-Color Policer Configuration Overview*
- *Hierarchical Policer Configuration Overview*
- *Guidelines for Applying Traffic Policers*

[edit protocols pppoe] Hierarchy Level

```
pppoe {
  no-send-pads-error;
  no-send-pads-ac-info
  pado-advertise;
  service-name-tables table-name {
    service service-name {
      drop;
      delay seconds;
      terminate;
      dynamic-profile profile-name;
      routing-instance routing-instance-name;
      max-sessions number;
      agent-specifier {
        aci circuit-id-string ari remote-id-string {
          drop;
          delay seconds;
          terminate;
          dynamic-profile profile-name;
          routing-instance routing-instance-name;
          static-interface interface-name;
        }
      }
    }
  }
  traceoptions {
    file <filename> <files number> <match regular-expression > <size maximum-file-size>
      <world-readable | no-world-readable>;
    filter {
      aci regular-expression;
      ari regular-expresion;
      service-name regular-expresion;
      underlying-interface interface-name;
    }
    flag flag;
    level (all | error | info | notice | verbose | warning);
    no-remote-trace;
  }
}
```

accept

Syntax accept (any | dhcp-v4 | dhcp-v6 | inet | inet6 | pppoe);

Hierarchy Level [edit interfaces *interface-name* auto-configure stacked-vlan-ranges dynamic-profile *profile-name*],
[edit interfaces *interface-name* auto-configure vlan-ranges dynamic-profile *profile-name*]

Release Information Statement introduced in Junos OS Release 9.5.
dhcp-v4 option added in Junos OS Release 10.0.
dhcp-v6, **inet6** and **pppoe** options added in Junos OS Release 10.2.
any option added in Junos OS Release 10.4.

Description Specify the type of VLAN Ethernet packet accepted by an interface that is associated with a VLAN dynamic profile or stacked VLAN dynamic profile.

Options **any**—Any packet type. Specifies that any incoming packets trigger the dynamic creation of a VLAN with properties determined by the auto-configure interface configuration stanza and associated profile attributes. This option is used when configuring wholesaling in a Layer 2 network.

dhcp-v4—IPv4 DHCP packet type. Specifies that incoming IPv4 DHCP discover packets trigger the dynamic creation of a VLAN with properties determined by the auto-configure interface configuration stanza and associated profile attributes

NOTE: The DHCP-specific **mac-address** and **option-82** options are rejected if the **accept** statement is not set to **dhcp-v4**.

dhcp-v6—IPv6 DHCP packet type. Specifies that incoming IPv6 DHCP discover packets trigger the dynamic creation of a VLAN with properties determined by the auto-configure interface configuration stanza and associated profile attributes.

inet—IPv4 Ethernet and ARP packet type.

inet6—IPv6 Ethernet packet type.

pppoe—Point-to-Point Protocol over Ethernet packet type.

NOTE: The **pppoe** VLAN Ethernet packet type option is supported only for Trio MPC/MIC interfaces on MX Series routers.

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

Related Documentation

- *Configuring the VLAN Ethernet Packet Type for Single-Tag VLAN Dynamic Profiles*
- *Configuring the VLAN Ethernet Packet Type for Stacked VLAN Dynamic Profiles*
- *Configuring VLAN Interfaces for the Layer 2 Wholesale Solution*
- Configuring Subscriber Packet Types to Trigger VLAN Authentication on page 22

access (Static Access Routes)

Syntax

```
access {
  route ip-prefix</prefix-length> {
    metric route-cost;
    next-hop next-hop;
    preference route-distance;
    qualified-next-hop next-hop;
    tag tag-number
  }
```

Hierarchy Level

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

Release Information

Statement introduced in Junos OS Release 10.1.
Statement introduced in Junos OS Release 12.3 for ACX Series routers.

Description

Configure access routes.

The remaining statements are explained separately.

Required Privilege Level

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

Related Documentation

- *Examples: Configuring Static Routes*

access-concentrator

Syntax access-concentrator *name*;

Hierarchy Level [edit dynamic-profiles *profile-name* interfaces demux0 unit *logical-unit-number* family pppoe],
[edit dynamic-profiles *profile-name* interfaces *interface-name* unit *logical-unit-number* family pppoe],
[edit interfaces *interface-name* unit *logical-unit-number* family pppoe],
[edit interfaces *interface-name* unit *logical-unit-number* pppoe-options],
[edit interfaces *interface-name* unit *logical-unit-number* pppoe-underlying-options],
[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number* family pppoe],
[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number* pppoe-options],
[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number* pppoe-underlying-options]

Release Information Statement introduced before Junos OS Release 7.4.
Support at the **[edit interfaces *interface-name* unit *logical-unit-number* pppoe-underlying-options]** and **[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number* pppoe-underlying-options]** hierarchy levels introduced in Junos OS Release 10.1.
Support at the **[edit ... family pppoe]** hierarchies introduced in Junos OS Release 11.2.

Description (Intelligent Queuing 2 (IQ2) PICs on M120 and M320 routers; MPCs on MX Series routers) Configure an alternative access concentrator name in the AC-NAME tag in a PPPoE control packet for use with a dynamic PPPoE subscriber interface. If you do not configure the access concentrator name, the AC-NAME tag contains the system name.

NOTE: The **[edit ... family pppoe]** hierarchies are supported only on MX Series routers with MPCs.

Options ***name***—Name of the access concentrator.

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

Related Documentation

- *Identifying the Access Concentrator*
- Configuring the PPPoE Family for an Underlying Interface on page 159
- Configuring Dynamic PPPoE Subscriber Interfaces Using Dynamic Profiles on page 145
- *PPPoE Overview*

access-profile

Syntax access-profile *name*;

Hierarchy Level
[edit interfaces *interface-name* auto-configure vlan-ranges],
[edit interfaces *interface-name* auto-configure stacked-vlan-ranges],
[edit interfaces *interface-name* ppp-options chap],
[edit interfaces *interface-name* ppp-options pap],
[edit interfaces *interface-name* unit *logical-unit-number* ppp-options chap],
[edit interfaces *interface-name* unit *logical-unit-number* ppp-options pap],
[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number* ppp-options chap],
[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number* ppp-options pap]

Release Information
Statement introduced before Junos OS Release 7.4.
Support for PAP added in Junos OS Release 8.3.
Support for VLAN and stacked VLAN ranges added in Junos OS Release 10.0.

Description
For CHAP authentication, the mapping between peer names (or “clients”) and the secrets associated with their respective links. For PAP authentication, the peer's username and password.

For Asynchronous Transfer Mode 2 (ATM2) IQ interfaces only, you can configure a Challenge Handshake Authentication Protocol (CHAP) access profile on the logical interface unit if the logical interface is configured with one of the following PPP over ATM encapsulation types:

- **atm-ppp-llc**—PPP over AAL5 logical link control (LLC) encapsulation.
- **atm-ppp-vc-mux**—PPP over AAL5 multiplex encapsulation.

For VLAN and stacked VLAN authentication, the access profile containing the RADIUS accounting and authentication information for the VLAN or stacked VLAN ranges.

Options *name*—Name of the access profile.

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

Related Documentation
- *Configuring the PPP Challenge Handshake Authentication Protocol*
- *Configuring the PPP Password Authentication Protocol*
- *default-chap-secret*
- *Junos OS Administration Library for Routing Devices*

address

Syntax address (*ip-address* | *ipv6-address*);

Hierarchy Level [edit dynamic-profiles *profile-name* interfaces *interface-name* unit *logical-unit-number* family *family*],
[edit dynamic-profiles *profile-name* interfaces demux0 unit *logical-unit-number* family *family*],
[edit dynamic-profiles *profile-name* interfaces pp0 unit "$junos-interface-unit" family *family*],
[edit interfaces *interface-name* unit *logical-unit-number* family inet],
[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number* family *family*]

Release Information Statement introduced in Junos OS Release 9.2.
Support at the **[edit dynamic-profiles *profile-name* interfaces pp0 unit "$junos-interface-unit" family *family*]** hierarchy level introduced in Junos OS Release 10.1.
Support at the **[edit interfaces *interface-name* unit *logical-unit-number* family *inet*]** hierarchy level introduced in Junos OS Release 13.2X50-D10 for EX Series switches.

Description Configure the interface address.

Options ***ip-address***—IPv4 address of the interface.

ipv6-address—IPv6 address of the interface. When configuring an IPv6 address on a dynamically created interface, use the ***$junos-ipv6–address*** dynamic variable.

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

Related Documentation
- *Configuring the Protocol Family*
- *Format for Specifying IP Addresses, Network Masks, and Prefixes in Junos OS Configuration Statements*

agent-circuit-identifier (Dynamic VLAN interface Sets)

Syntax

```
agent-circuit-identifier {
  dynamic-profile profile-name;
}
```

Hierarchy Level

[edit dynamic-profiles *profile-name* interfaces "$junos-interface-ifd-name" unit "$junos-interface-unit" auto-configure],
[edit interfaces *interface-name* unit *logical-unit-number* auto-configure]

Release Information

Statement introduced in Junos OS Release 12.2.

Description

Configure a static or dynamic underlying VLAN interface to enable dynamic VLAN subscriber interface creation based on agent circuit identifier information.

The remaining statements are explained separately.

Required Privilege Level

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

Related Documentation

- Configuring Dynamic Underlying VLAN Interfaces to Use Agent Circuit Identifier Information on page 33
- Configuring Static Underlying VLAN Interfaces to Use Agent Circuit Identifier Information on page 35

aggregate-clients (DHCP Local Server)

Syntax

aggregate-clients (merge | replace);

Hierarchy Level

[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* system services dhcp-local-server dynamic-profile *profile-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* system services dhcp-local-server group *group-name* dynamic-profile *profile-name*],
[edit logical-systems *logical-system-name* system services dhcp-local-server dynamic-profile *profile-name*],
[edit logical-systems *logical-system-name* system services dhcp-local-server group *group-name* dynamic-profile *profile-name*],
[edit routing-instances *routing-instance-name* system services dhcp-local-server dynamic-profile *profile-name*],
[edit routing-instances *routing-instance-name* system services dhcp-local-server group *group-name* dynamic-profile *profile-name*],
[edit system services dhcp-local-server dynamic-profile *profile-name*],
[edit system services dhcp-local-server group *group-name* dynamic-profile *profile-name*]

Release Information

Statement introduced in Junos OS Release 9.3.
Options **merge** and **replace** introduced in Junos OS Release 9.5.

Description

Specify that the router merge (chain) client attributes such as firewall filters and CoS attributes or replace them when multiple client sessions exist on the same underlying VLAN.

Not supported for IP demux subscriber interfaces.

Options

merge—Aggregate multiple clients attributes for the same subscriber (logical interface)

replace—Replace the entire logical interface whenever a new client logs in to the network using the same VLAN logical interface

Required Privilege Level

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

Related Documentation

- Attaching Dynamic Profiles to DHCP Subscriber Interfaces or DHCP Client Interfaces on page 114

anchor-point (Pseudowire Subscriber Interfaces)

Syntax anchor-point *lt-device*;

Hierarchy Level [edit interfaces ps0]

Release Information Statement introduced in Junos OS Release 13.1.

Description Specify the logical tunnel (lt) interface that identifies the Packet Forwarding Engine that processes the pseudowire termination.

Options *lt-device*—An lt device in the format **lt-*fpc/pic/port***

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

Related Documentation

- Pseudowire Subscriber Logical Interfaces Overview on page 321
- Configuring a Pseudowire Subscriber Logical Interface on page 323
- Configuring a Pseudowire Subscriber Logical Interface Device on page 325

arp (Interfaces)

Syntax arp *ip-address* (mac | multicast-mac) *mac-address* publish;

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

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.
Statement introduced in Junos OS Release 11.1 for the QFX Series.

Description For Ethernet, Fast Ethernet, and Gigabit Ethernet interfaces only, configure Address Resolution Protocol (ARP) table entries, mapping IP addresses to MAC addresses.

Options *ip-address*—IP address to map to the MAC address. The IP address specified must be part of the subnet defined in the enclosing **address** statement.

mac *mac-address*—MAC address to map to the IP address. Specify the MAC address as six hexadecimal bytes in one of the following formats: ***nnnn.nnnn.nnnn*** or ***nn*:*nn*:*nn*:*nn*:*nn*:*nn***. For example, **0011.2233.4455** or **00:11:22:33:44:55**.

multicast-mac *mac-address*—Multicast MAC address to map to the IP address. Specify the multicast MAC address as six hexadecimal bytes in one of the following formats: ***nnnn.nnnn.nnnn*** or ***nn*:*nn*:*nn*:*nn*:*nn*:*nn***. For example, **0011.2233.4455** or **00:11:22:33:44:55**.

publish—(Optional) Have the router or switch reply to ARP requests for the specified IP address. If you omit this option, the router or switch uses the entry to reach the destination but does not reply to ARP requests.

NOTE: The edit logical-systems hierarchy is not available on QFabric systems.

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

Related Documentation
- *Configuring Static ARP Table Entries*

atm-options

Syntax

```
atm-options {
  cell-bundle-size cells;
  ilmi;
  linear-red-profiles profile-name {
    high-plp-max-threshold percent;
    low-plp-max-threshold percent;
    queue-depth cells high-plp-threshold percent low-plp-threshold percent;
  }
  mpls {
    pop-all-labels {
      required-depth number;
    }
  }
  pic-type (atm1 | atm2);
  plp-to-clp;
  promiscuous-mode {
    vpi vpi-identifier;
  }
  scheduler-maps map-name {
    forwarding-class class-name {
      epd-threshold cells plp1 cells;
      linear-red-profile profile-name;
      priority (high | low);
      transmit-weight (cells number | percent number);
    }
    vc-cos-mode (alternate | strict);
  }
  use-null-cw;
  vpi vpi-identifier {
    maximum-vcs maximum-vcs;
    oam-liveness {
      up-count cells;
      down-count cells;
    }
    oam-period (disable | seconds);
    shaping {
      (cbr rate | rtvbr peak rate sustained rate burst length | vbr peak rate sustained rate burst
        length);
      queue-length number;
    }
  }
}
```

Hierarchy Level [edit interfaces *interface-name*]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 12.2 for the ACX Series Universal Access Routers.

Description Configure ATM-specific physical interface properties.

The statements are explained separately.

NOTE: Certain options apply only to specific platforms.

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

Related Documentation
- *Interface Encapsulations Overview*
- *multipoint-destination*
- *shaping*
- vci on page 592

authentication

Syntax

```
authentication {
  packet-types [packet-types];
  password password-string;
  username-include {
    circuit-type;
    delimiter delimiter-character;
    domain-name domain-name-string;
    interface-name;
    mac-address;
    option-82 <circuit-id> <remote-id>;
    radius-realm radius-realm-string;
    user-prefix user-prefix-string;
  }
}
```

Hierarchy Level
[edit interfaces *interface-name* auto-configure vlan-ranges],
[edit interfaces *interface-name* auto-configure stacked-vlan-ranges]

Release Information
Statement introduced in Junos OS Release 10.0.

Description
Specify the authentication parameters that trigger the Access-Request message to AAA for the interface.

The remaining statements are explained separately.

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

Related Documentation
- *Configuring Subscribers over Static Interfaces*
- *Configuring the Static Subscriber Global Authentication Password*

auto-configure

Syntax

```
auto-configure {
  vlan-ranges {
    access-profile profile-name;
    authentication {
      packet-types [packet-types];
      password password-string;
      username-include{
        circuit-type;
        delimiter delimiter-character;
        domain-name domain-name-string;
        interface-name;
        mac-address;
        option-18;
        option-37;
        option-82 <circuit-id> <remote-id>;
        radius-realm radius-realm-string;
        user-prefix user-prefix-string;
      }
    }
    dynamic-profile profile-name {
      accept (any | dhcp-v4 |dhcp-v6| inet | inet6 | pppoe);
      ranges (any | low-tag)–(any | high-tag);
    }
    override;
  }
  stacked-vlan-ranges {
    access-profile profile-name;
    authentication {
      packet-types [packet-types];
      password password-string;
      username-include {
        circuit-type;
        delimiter delimiter-character;
        domain-name domain-name-string;
        interface-name;
        mac-address;
        option-18;
        option-37;
        option-82 <circuit-id> <remote-id>;
        radius-realm radius-realm-string;
        user-prefix user-prefix-string;
      }
    }
    dynamic-profile profile-name {
      accept (any | dhcp-v4 |dhcp-v6| inet | inet6 | pppoe);
      ranges (any | low-tag–high-tag),(any | low-tag–high-tag);
    }
    override;
  }
  remove-when-no-subscribers;
}
```


Hierarchy Level [edit interfaces *interface-name*]

Release Information Statement introduced in Junos OS Release 9.5.

Description Enable the configuration of dynamic, auto-sensed VLANs.

The remaining statements are explained separately.

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

Related Documentation

- *Configuring VLAN Interfaces to Use Dynamic Profiles*

auto-configure (Dynamic VLAN Interface Sets)

Syntax

```
auto-configure {
  agent-circuit-identifier {
    dynamic-profile profile-name;
  }
}
```

Hierarchy Level [edit dynamic-profiles *profile-name* interfaces “$junos-interface-ifd-name” unit “$junos-interface-unit”],
[edit interfaces *interface-name* unit *logical-unit-number*]

Release Information Statement introduced in Junos OS Release 12.2.

Description Enable the configuration of dynamic, auto-sensed VLAN subscriber interfaces on a static or dynamic underlying VLAN interface.

The remaining statements are explained separately.

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

Related Documentation

- Configuring Dynamic Underlying VLAN Interfaces to Use Agent Circuit Identifier Information on page 33
- Configuring Static Underlying VLAN Interfaces to Use Agent Circuit Identifier Information on page 35

chap

Syntax

```
chap {
  access-profile name;
  challenge-length minimum minimum-length maximum maximum-length;
  default-chap-secret name;
  local-name name;
  passive;
}
```

Hierarchy Level

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

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Allow each side of a link to challenge its peer, using a “secret” known only to the authenticator and that peer. The secret is not sent over the link.

By default, PPP CHAP is disabled. If CHAP is not explicitly enabled, the interface makes no CHAP challenges and denies all incoming CHAP challenges.

For ATM2 IQ interfaces only, you can configure CHAP on the logical interface unit if the logical interface is configured with one of the following PPP over ATM encapsulation types:

- **atm-ppp-llc**—PPP over AAL5 LLC encapsulation.
- **atm-ppp-vc-mux**—PPP over AAL5 multiplex encapsulation.

BEST PRACTICE: On inline service (si) interfaces for L2TP, only the **chap** statement itself is typically used for subscriber management. We recommend that you leave the subordinate statements at their default values.

The remaining statements are explained separately.

Required Privilege Level

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

Related Documentation

- *Configuring the PPP Challenge Handshake Authentication Protocol*
- *Applying PPP Attributes to L2TP LNS Subscribers with a User Group Profile*
- *Applying PPP Attributes to L2TP LNS Subscribers Per Inline Service Interface*

chap (Dynamic PPP)

Syntax

```
chap {
  challenge-length minimum minimum-length maximum maximum-length;
}
```

Hierarchy Level

[edit dynamic-profiles *profile-name* interfaces pp0 unit “$junos-interface-unit” ppp-options],
[edit dynamic-profiles *profile-name* interfaces "$junos-interface-ifd-name" unit
“$junos-interface-unit” ppp-options]

Release Information

Statement introduced in Junos OS Release 9.5.
Support at the **[edit dynamic-profiles *profile-name* interfaces "$junos-interface-ifd-name" unit “$junos-interface-unit” ppp-options]** hierarchy level introduced in Junos OS Release 12.2.

Description

Specify CHAP authentication in a PPP dynamic profile.

The remaining statement is explained separately.

Required Privilege Level

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

Related Documentation

- *Dynamic Profiles Overview*
- *Configuring Dynamic Authentication for PPP Subscribers*
- *Attaching Dynamic Profiles to Static PPP Subscriber Interfaces*
- *Applying PPP Attributes to L2TP LNS Subscribers Per Inline Service Interface*

circuit-type

Syntax

```
circuit-type;
```

Hierarchy Level

[edit interfaces *interface-name* auto-configure vlan-ranges authentication username-include],
[edit interfaces *interface-name* auto-configure stacked-vlan-ranges authentication
username-include],

Release Information

Statement introduced in Junos OS Release 10.0.

Description

Specify that the circuit type is concatenated with the username during the subscriber authentication process.

Required Privilege Level

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

Related Documentation

- Configuring VLAN Interface Username Information for AAA Authentication on page 22

class-of-service (Dynamic Profiles)

Syntax class-of-service { ... }

Hierarchy Level [edit dynamic-profiles *profile-name*]

Release Information Statement introduced in Junos OS Release 9.2.

Description Configure Junos OS CoS features in a dynamic profile.

Default If you do not configure any CoS features, all packets are transmitted from output transmission queue 0.

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

Related Documentation

- *Guidelines for Configuring Dynamic CoS for Subscriber Access*
- *Configuring Static Hierarchical Scheduling and Queuing in a Dynamic Profile for Subscriber Access*
- *Configuring Dynamic Hierarchical Scheduling and Queuing in a Dynamic Profile for Subscriber Access*

delimiter

Syntax delimiter *delimiter-character*;

Hierarchy Level [edit interfaces *interface-name* auto-configure vlan-ranges authentication username-include],
[edit interfaces *interface-name* auto-configure stacked-vlan-ranges authentication username-include]

Release Information Statement introduced in Junos OS Release 10.0.

Description Specify the character used as the delimiter between the concatenated components of the username. You cannot use the semicolon (;) as a delimiter.

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

Related Documentation

- Configuring VLAN Interface Username Information for AAA Authentication on page 22

demux-options (Dynamic Interface)

Syntax

```
demux–options {
  underlying-interface interface-name
}
```

Hierarchy Level [edit dynamic-profiles *profile-name* interfaces demux0 *interface-name* unit *logical-unit-number*]

Release Information Statement introduced in Junos OS Release 9.3.

Description Configure logical demultiplexing (demux) interface options in a dynamic profile.

The remaining statement is explained separately.

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

Related Documentation

- Configuring Dynamic Subscriber Interfaces Using IP Demux Interfaces in Dynamic Profiles on page 77
- *Demultiplexing Interface Overview*

demux-source (Dynamic IP Demux Interface)

Syntax

```
demux-source {
  source-address;
}
```

Hierarchy Level [edit dynamic-profiles *profile-name* interfaces demux0 unit *logical-unit-number* family *family*]

Release Information Statement introduced in Junos OS Release 9.3.

Description Configure a logical demultiplexing (demux) source address for a subscriber in a dynamic profile.

Options ***source-address***—Either the specific source address you want to assign to the subscriber interface or the source address variable. For IPv4, specify **$junos-subscriber-ip-address**; for IPv6, specify **$junos-subscriber-ipv6-address**). The source address for the interface is dynamically supplied by DHCP when the subscriber accesses the router.

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

Related Documentation

- Configuring Dynamic Subscriber Interfaces Using IP Demux Interfaces in Dynamic Profiles on page 77
- *Demultiplexing Interface Overview*

demux-source (Dynamic Underlying Interface)

Syntax demux-source *family*;

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

Release Information Statement introduced in Junos OS Release 9.6.

Description Configure the logical demultiplexing (demux) source family type on the IP demux underlying interface within a dynamic profile.

NOTE: **The IP demux interface feature currently supports only Fast Ethernet, Gigabit Ethernet, 10-Gigabit Ethernet, or aggregated Ethernet underlying interfaces.**

Options ***family***—Protocol family:

- **inet**—Internet Protocol version 4 suite
- **inet6**—Internet Protocol version 6 suite

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

demux0 (Dynamic Interface)

Syntax

```
demux0 {
  unit logical-unit-number {
    demux-options {
      underlying-interface interface-name
    }
    family family {
      access-concentrator name;
      address address;
      demux-source {
        source-prefix;
      }
      direct-connect;
      duplicate-protection;
      dynamic-profile profile-name;
      filter {
        input filter-name;
        output filter-name;
      }
      mac-validate (loose | strict):
      max-sessions number;
      max-sessions-vsa-ignore;
      rpf-check {
        fail-filter filter-name;
        mode loose;
      }
      service-name-table table-name
      short-cycle-protection <lockout-time-min minimum-seconds lockout-time-max
        maximum-seconds>;
      unnumbered-address interface-name <preferred-source-address address>;
    }
    filter {
      input filter-name;
      output filter-name;
    }
    vlan-id number;
  }
}
```

Hierarchy Level [edit dynamic-profiles *profile-name* interfaces]

Release Information Statement introduced in Junos OS Release 9.3.

Description Configure the logical demultiplexing (demux) interface in a dynamic profile.

Logical IP demux interfaces do not support IPv4 and IPv6 dual stack.

The remaining statements are explained separately.

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

Related Documentation

- Configuring Dynamic Subscriber Interfaces Using IP Demux Interfaces in Dynamic Profiles on page 77
- *Demultiplexing Interface Overview*

destination (Tunnels)

Syntax destination *address*;

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

Release Information
Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 12.1 for EX Series switches.
Statement introduced in Junos OS Release 13.2 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description For encrypted, PPP-encapsulated, and tunnel interfaces, specify the remote address of the connection.

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

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

Related Documentation

- *Configuring the Interface Address*
- *point-to-point*

device-count (Pseudowire Subscriber Interfaces)

Syntax device-count *number*;

Hierarchy Level [edit chassis pseudowire-service]

Release Information Statement introduced in Junos OS Release 13.1.

Description Configure the number of pseudowire logical devices available to the router.

Options *number*—Number of devices.

Range: 1 through 2048

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

Related Documentation

- Pseudowire Subscriber Logical Interfaces Overview on page 321
- Configuring a Pseudowire Subscriber Logical Interface on page 323
- Configuring the Maximum Number of Pseudowire Logical Interface Devices Supported on the Router on page 325

direct-connect

Syntax direct-connect;

Hierarchy Level
[edit dynamic-profiles *profile-name* interfaces demux0 unit *logical-unit-number* family pppoe],
[edit dynamic-profiles *profile-name* interfaces *interface-name* unit *logical-unit-number* family pppoe],
[edit dynamic-profiles *profile-name* interfaces *interface-name* unit *logical-unit-number* pppoe-underlying-options],
[edit interfaces *interface-name* unit *logical-unit-number*family pppoe],
[edit interfaces *interface-name* unit *logical-unit-number* pppoe-underlying-options],
[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number* family pppoe],
[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number* pppoe-underlying-options]

Release Information Statement introduced in Junos OS 13.3.

Description Configure the router to ignore any DSL Forum VSAs that it receives in PPPoE control packets when the router is directly connected to CPE devices.

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

Related Documentation

- Ignoring DSL Forum VSAs from Directly Connected Devices on page 161
- Configuring an Underlying Interface for Dynamic PPPoE Subscriber Interfaces on page 156
- Configuring the PPPoE Family for an Underlying Interface on page 159
- Dynamic PPPoE Subscriber Interfaces over Static Underlying Interfaces Overview on page 153

domain-name

Syntax domain-name *domain-name-string*;

Hierarchy Level [edit forwarding-options dhcp-relay authentication username-include],
[edit forwarding-options dhcp-relay group *group-name* authentication username-include],
[edit logical-systems *logical-system-name* forwarding-options dhcp-relay authentication username-include],
[edit logical-systems *logical-system-name* forwarding-options dhcp-relay group *group-name* authentication username-include],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* forwarding-options dhcp-relay authentication username-include],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* forwarding-options dhcp-relay group *group-name* authentication username-include],
[edit routing-instances *routing-instance-name* forwarding-options dhcp-relay authentication username-include],
[edit routing-instances *routing-instance-name* forwarding-options dhcp-relay group *group-name* authentication username-include]

Release Information Statement introduced in Junos OS Release 9.1.

Description Specify the domain name that is concatenated with the username during the subscriber authentication process.

Options ***domain-name-string***—The domain name formatted string.

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

Related Documentation
- *Using External AAA Authentication Services with DHCP*

duplicate-protection (Dynamic PPPoE)

Syntax duplicate-protection;

Hierarchy Level [edit dynamic-profiles *profile-name* interfaces demux0 unit *logical-unit-number* family pppoe],
[edit dynamic-profiles *profile-name* interfaces *interface-name* unit *logical-unit-number* family pppoe],
[edit interfaces *interface-name* unit *logical-unit-number*family pppoe],
[edit interfaces *interface-name* unit *logical-unit-number* pppoe-underlying-options],
[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number* family pppoe],
[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number* pppoe-underlying-options]

Release Information Statement introduced in Junos OS Release 10.1.
Support for the **[edit ... family pppoe]** hierarchies introduced in Junos OS Release 11.2.

Description Prevent the activation of another dynamic PPPoE logical interface on the same underlying interface when a dynamic PPPoE logical interface for a client with the same media access control (MAC) address is already active on that interface. Duplicate protection is disabled by default. Enabling duplicate protection has no effect on dynamic PPPoE logical interfaces that are already active.

NOTE: The **[edit ... family pppoe]** hierarchies are supported only on MX Series routers with MPCs.

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

Related Documentation

- Configuring an Underlying Interface for Dynamic PPPoE Subscriber Interfaces on page 156
- Configuring the PPPoE Family for an Underlying Interface on page 159
- Configuring Lockout of PPPoE Subscriber Sessions on page 198
- Dynamic PPPoE Subscriber Interfaces over Static Underlying Interfaces Overview on page 153

dynamic-profile (Dynamic VLAN Interface Sets)

Syntax dynamic-profile *profile-name*;

Hierarchy Level [edit dynamic-profiles *profile-name* interfaces "$junos-interface-ifd-name" unit "$junos-interface-unit" auto-configure agent-circuit-identifier],
[edit interfaces *interface-name* unit *logical-unit-number* auto-configure agent-circuit-identifier]

Release Information Statement introduced in Junos OS Release 12.2.

Description Attach a dynamic profile for an agent circuit identifier (ACI) interface set to a static or dynamic underlying VLAN interface.

Options

- *profile-name*—Name of the dynamic profile that defines the ACI interface set.

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

Related Documentation

- Configuring Dynamic Underlying VLAN Interfaces to Use Agent Circuit Identifier Information on page 33
- Configuring Static Underlying VLAN Interfaces to Use Agent Circuit Identifier Information on page 35

dynamic-profile (Dynamic PPPoE)

Syntax dynamic-profile *profile-name*;

Hierarchy Level [edit dynamic-profiles *profile-name* interfaces demux0 unit *logical-unit-number* family pppoe],
[edit dynamic-profiles *profile-name* interfaces *interface-name* unit *logical-unit-number* family pppoe],
[edit interfaces *interface-name* unit *logical-unit-number* family pppoe],
[edit interfaces *interface-name* unit *logical-unit-number* pppoe-underlying-options],
[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number* family pppoe],
[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number* pppoe-underlying-options]

Release Information Statement introduced in Junos OS Release 10.1.
Support for the **[edit ... family pppoe]** hierarchies introduced in Junos OS Release 11.2.

Description Attach a PPPoE dynamic profile to an underlying Ethernet interface. This underlying interface is configured with either the **encapsulation ppp-over-ether** statement or the **family pppoe** statement; the two statements are mutually exclusive. When the router creates a dynamic PPPoE logical interface on the underlying interface, it uses the information in the dynamic profile to determine the properties of the dynamic PPPoE logical interface.

NOTE: The **[edit ... family pppoe]** hierarchies are supported only on MX Series routers with MPCs.

Options *profile-name*—Name of a previously configured PPPoE dynamic profile, up to 64 characters in length, defined at the **[edit dynamic-profiles *profile-name* interfaces pp0]** hierarchy level.

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

Related Documentation

- Configuring an Underlying Interface for Dynamic PPPoE Subscriber Interfaces on page 156
- Configuring the PPPoE Family for an Underlying Interface on page 159
- Dynamic PPPoE Subscriber Interfaces over Static Underlying Interfaces Overview on page 153

dynamic-profile (PPP)

Syntax dynamic-profile *profile-name*;

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

Release Information Statement introduced in Junos OS Release 9.5.
Support for MLPPP on LSQ interfaces introduced in Junos OS Release 10.2.

Description Specify the dynamic profile that is attached to the interface. On the MX Series routers, this statement is currently supported on PPPoE interfaces only. On the M120 and M320 routers, this statement is supported for MLPPP bundles only on LSQ interfaces on Adaptive Services PICs and Multiservices PICs.

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

Related Documentation

- *Dynamic Profiles Overview*
- *Configuring a Basic Dynamic Profile*
- *Attaching Dynamic Profiles to Static PPP Subscriber Interfaces*
- Attaching Dynamic Profiles to MLPPP Bundles on page 307
- For hardware requirements, see Hardware Requirements for PPP Subscriber Services on Non-Ethernet Interfaces on page 306

dynamic-profile (PPPoE Service Name Tables)

Syntax dynamic-profile *profile-name*;

Hierarchy Level [edit protocols pppoe service-name-tables *table-name* service *service-name*],
[edit protocols pppoe service-name-tables *table-name* service *service-name* agent-specifier aci *circuit-id-string* ari *remote-id-string*]

Release Information Statement introduced in Junos OS Release 10.2.

Description Specify a dynamic profile to instantiate a dynamic PPPoE interface. You can associate a dynamic profile with a named service entry, **empty** service entry, or **any** service entry configured in a PPPoE service name table, or with an agent circuit identifier/agent remote identifier (ACI/ARI) pair defined for these services.

The dynamic profile associated with a service entry in a PPPoE service name table overrides the dynamic profile associated with the PPPoE underlying interface on which the dynamic PPPoE interface is created.

If you include the **dynamic-profile** statement at the **[edit protocols pppoe service-name-tables *table-name* service *service-name* agent-specifier aci *circuit-id-string* ari *remote-id-string*]** hierarchy level, you cannot also include the **static-interface** statement at this level. The **dynamic-profile** and **static-interface** statements are mutually exclusive for ACI/ARI pair configurations.

Options *profile-name*—Name of the dynamic profile that the router uses to instantiate a dynamic PPPoE interface.

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

Related Documentation

- *Configuring PPPoE Service Name Tables*
- Assigning a Dynamic Profile and Routing Instance to a Service Name or ACI/ARI Pair for Dynamic PPPoE Interface Creation on page 150

dynamic-profile (Stacked VLAN)

Syntax

```
dynamic-profile profile-name {
  accept (any | dhcp-v4 |dhcp-v6| inet | inet6 | pppoe);
  ranges (any | low-tag–high-tag) ,(any | low-tag–high-tag);
}
```

Hierarchy Level [edit interfaces *interface-name* auto-configure stacked-vlan-ranges]

Release Information Statement introduced in Junos OS Release 9.5.

Description Configure a dynamic profile for use when configuring dynamic stacked VLANs.

Options ***profile-name***—Name of the dynamic profile that you want to use when configuring dynamic stacked VLANs.

The remaining statements are explained separately.

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

Related Documentation

- *Dynamic Profiles Overview*
- *Configuring a Basic Dynamic Profile*
- *Associating a Stacked VLAN Dynamic Profile with an Interface*

dynamic-profile (VLAN)

Syntax

```
dynamic-profile profile-name {
  accept (any | dhcp-v4 |dhcp-v6| inet | inet6 | pppoe);
  ranges (any | low-tag)–(any | high-tag);
}
```

Hierarchy Level [edit interfaces *interface-name* auto-configure vlan-ranges]

Release Information Statement introduced in Junos OS Release 9.5.

Description Configure a dynamic profile for use when configuring dynamic VLANs.

Options ***profile-name***—Name of the dynamic profile that you want to use when configuring dynamic VLANs.

The remaining statements are explained separately.

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

Related Documentation

- *Dynamic Profiles Overview*
- *Configuring a Basic Dynamic Profile*
- *Associating a Single-Tag VLAN Dynamic Profile with an Interface*

dynamic-profiles

Syntax

```
dynamic-profiles {
  profile-name {
    class-of-service {
      interfaces {
        interface-name ;
        }
        unit logical-unit-number {
          classifiers {
            type (classifier-name | default);
          }
          output-traffic-control-profile (profile-name | $junos-cos-traffic-control-profile);
          rewrite-rules {
            dscp (rewrite-name | default);
            dscp-ipv6 (rewrite-name | default);
            ieee-802.1 (rewrite-name | default) vlan-tag (outer | outer-and-inner);
            inet-precedence (rewrite-name | default);
            }
          }
        }
      }
      scheduler-maps {
        map-name {
          forwarding-class class-name scheduler scheduler-name;
        }
      }
      schedulers {
        (scheduler-name) {
          buffer-size (seconds | percent percentage | remainder | temporal microseconds);
          drop-profile-map loss-priority (any | low | medium-low | medium-high | high)
            protocol (any | non-tcp | tcp) drop-profile profile-name;
          excess-priority (low | high | $junos-cos-scheduler-excess-priority);
          excess-rate (percent percentage | percent $junos-cos-scheduler-excess-rate);
          overhead-accounting (shaping-mode) <bytes (byte-value>;
          priority priority-level;
          shaping-rate (rate | predefined-variable);
          transmit-rate (percent percentage | rate | remainder) <exact | rate-limit>;
        }
      }
      traffic-control-profiles profile-name {
        delay-buffer-rate (percent percentage | rate | $junos-cos-delay-buffer-rate);
        excess-rate (percent percentage | proportion value | percent $junos-cos-excess-rate);
        guaranteed-rate (percent percentage | rate | $junos-cos-guaranteed-rate);
        overhead-accounting (shaping-mode) <bytes (byte-value>;
        scheduler-map map-name;
        shaping-rate (rate | predefined-variable);
      }
    }
    firewall {
      family family {
        fast-update-filter filter-name {
          interface-specific;
          match-order [match-order];
```



```
    term term-name {
      from {
        match-conditions;
      }
      then {
        action;
        action-modifiers;
      }
      only-at-create;
    }
  }
  filter uid {
    enhanced-mode-override;
    interface-shared;
    interface-specific;
    term term-name {
      from {
        match-conditions;
      }
      then {
        action;
        action-modifiers;
      }
    }
  }
}
policer uid {
  filter-specific;
  if-exceeding {
    (bandwidth-limit bps | bandwidth-percent percentage);
    burst-size-limit bytes;
  }
  logical-bandwidth-policer;
  logical-interface-policer;
  physical-interface-policer;
  then {
    policer-action;
  }
}
hierarchical-policer uid {
  aggregate {
    if-exceeding {
      bandwidth-limit-limit bps;
      burst-size-limit bytes;
    }
    then {
      policer-action;
    }
  }
  premium {
    if-exceeding {
      bandwidth-limit bps;
      burst-size-limit bytes;
    }
    then {
      policer-action;
```

```
            }
        }
    }
    three-color-policer uid {
        action {
            loss-priority high then discard;
        }
        logical-interface-policer;
        single-rate {
            (color-aware | color-blind);
            committed-burst-size bytes;
            committed-information-rate bps;
            excess-burst-size bytes;
        }
        two-rate {
            (color-aware | color-blind);
            committed-burst-size bytes;
            committed-information-rate bps;
            peak-burst-size bytes;
            peak-information-rate bps;
            }
        }
    }
}
policy-options {
    prefix-list uid {
        ip-addresses;
        dynamic-db;
    }
}
interfaces interface-name {
    interface-set interface-set-name {
        interface interface-name {
            unit logical unit number {
                advisory-options {
                    downstream-rate rate;
                    upstream-rate rate;
                }
            }
        }
    }
    unit logical-unit-number {
        auto-configure {
            agent-circuit-identifier {
                dynamic-profile profile-name;
            }
        }
        encapsulation (atm-ccc-cell-relay | atm-ccc-vc-mux | atm-cisco-nlpid |
            atm-tcc-vc-mux | atm-mlppp-llc | atm-nlpid | atm-ppp-llc | atm-ppp-vc-mux |
            atm-snap | atm-tcc-snap | atm-vc-mux | ether-over-atm-llc |
            ether-vpls-over-atm-llc | ether-vpls-over-fr | ether-vpls-over-ppp | ethernet |
            frame-relay-ccc | frame-relay-ppp | frame-relay-tcc | frame-relay-ether-type |
            frame-relay-ether-type-tcc | multilink-frame-relay-end-to-end | multilink-ppp |
            ppp-over-ether | ppp-over-ether-over-atm-llc | vlan-bridge | vlan-ccc | vlan-vci-ccc
            | vlan-tcc | vlan-vpls);
        family family {
```



```
        address address;
        filter {
          adf {
            counter;
            input-precedence precedence;
            not-mandatory;
            output-precedence precedence;
            rule rule-value;
          }
          input filter-name (
            precedence precedence;
          }
          output filter-name {
            precedence precedence;
          }
        }
        rpf-check {
          fail-filter filter-name;
          mode loose;
        }
        service {
          input {
            service-set service-set-name {
              service-filter filter-name;
            }
            post-service-filter filter-name;
          }
          input-vlan-map {
            inner-tag-protocol-id tpid;
            inner-vlan-id number;
            (push | swap);
            tag-protocol-id tpid;
            vlan-id number;
          }
          output {
            service-set service-set-name {
              service-filter filter-name;
            }
          }
          output-vlan-map {
            inner-tag-protocol-id tpid;
            inner-vlan-id number;
            (pop | swap);
            tag-protocol-id tpid;
            vlan-id number;
          }
        }
        unnumbered-address interface-name <preferred-source-address address>;
      }
      ppp-options {
        chap;
        pap;
      }
      vlan-id number;
      vlan-tags outer [tpid].vlan-id [inner [tpid].vlan-id];
    }
```

```
}
interfaces {
  demux0 {...}
}
interfaces {
  pp0 {...}
}
protocols {
  igmp {
    interface interface-name {
      accounting;
      disable;
      group-policy;
      immediate-leave
      no-accounting;
      promiscuous-mode;
      ssm-map ssm-map-name;
      static {
        group group {
          source source;
        }
      }
      version version;
    }
    mld {
      interfaceinterface-name {
        disable;
        (accounting | no-accounting);
        group-policy;
        immediate-leave;
        oif-map;
        passive;
        ssm-map ssm-map-name;
        static {
          group multicast-group-address {
            exclude;
            group-count number;
            group-increment increment;
            source ip-address {
              source-count number;
              source-increment increment;
            }
          }
        }
        version version;
      }
    }
    router-advertisement {
      interface interface-name {
        current-hop-limit number;
        default-lifetime seconds;
        (managed-configuration | no-managed-configuration);
        max-advertisement-interval seconds;
        min-advertisement-interval seconds;
        (other-stateful-configuration | no-other-stateful-configuration);
        prefix prefix;
```



```
            reachable-time milliseconds;
            retransmit-timer milliseconds;
          }
        }
      }
    }
    routing-instances routing-instance-name {
      interface interface-name;
      routing-options {
        access {
          route prefix {
            next-hop next-hop;
            metric route-cost;
            preference route-distance;
            tag route-tag;
          }
        }
        access-internal {
          route subscriber-ip-address {
            qualified-next-hop underlying-interface {
              mac-address address;
            }
          }
        }
        multicast {
          interface interface-name {
            no-qos-adjust;
          }
        }
      }
      rib routing-table-name {
        access {
          route prefix {
            next-hop next-hop;
            metric route-cost;
            preference route-distance;
            tag route-tag;
          }
        }
        access-internal {
          route subscriber-ip-address {
            qualified-next-hop underlying-interface {
              mac-address address;
            }
          }
        }
      }
    }
    routing-options {
      access {
        route prefix {
          next-hop next-hop;
          metric route-cost;
          preference route-distance;
          tag route-tag;
        }
```

```
                }
                access-internal {
                    route subscriber-ip-address {
                        qualified-next-hop underlying-interface {
                            mac-address address;
                        }
                    }
                }
                multicast {
                    interface interface-name {
                        no-qos-adjust;
                    }
                }
            }
            variables {
                variable-name {
                    default-value default-value;
                    equals expression;
                    mandatory;
                    uid;
                    uid-reference;
                }
            }
        }
    }
```

Hierarchy Level [edit]

Release Information Statement introduced in Junos OS Release 9.2.
Support at the **filter, policer, hierarchical-policer, three-color-policer, and policy options** hierarchy levels introduced in Junos OS Release 11.4.

Description Create dynamic profiles for use with DHCP or PPP client access.

Options ***profile-name***—Name of the dynamic profile; string of up to 80 alphanumeric characters.

The remaining statements are explained separately.

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

Related Documentation

- *Configuring a Basic Dynamic Profile*
- Configuring Dynamic VLANs Based on Agent Circuit Identifier Information on page 29
- *Dynamic Profiles Overview*

encapsulation (Logical Interface)

Syntax

encapsulation (atm-ccc-cell-relay | atm-ccc-vc-mux | atm-cisco-nlpid | atm-mlppp-llc | atm-nlpid | atm-ppp-llc | atm-ppp-vc-mux | atm-snap | atm-tcc-snap | atm-tcc-vc-mux | atm-vc-mux | ether-over-atm-llc | ether-vpls-over-atm-llc | ether-vpls-over-fr | ether-vpls-over-ppp | ethernet | ethernet-ccc | ethernet-vpls | ethernet-vpls-fr | frame-relay-ccc | frame-relay-ether-type | frame-relay-ether-type-tcc | frame-relay-ppp | frame-relay-tcc | gre-fragmentation | multilink-frame-relay-end-to-end | multilink-ppp | ppp-over-ether | ppp-over-ether-over-atm-llc | vlan-bridge | vlan-ccc | vlan-vci-ccc | vlan-tcc | vlan-vpls | vxlan);

Hierarchy Level

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

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 12.1X48 for PTX Series Packet Transport Routers (**ethernet**,**vlan-ccc**, and **vlan-tcc** options only).
Statement introduced in Junos OS Release 12.2 for the ACX Series Universal Access Routers. Only the **atm-ccc-cell-relay** and **atm-ccc-vc-mux** options are supported on ACX Series routers.
Support for **vlan-bridge** option introduced in Junos OS Release 14.1X53-D35 for the QFX Series.

Description

Configure a logical link-layer encapsulation type.

Options

atm-ccc-cell-relay—Use ATM cell-relay encapsulation.

atm-ccc-vc-mux—Use ATM virtual circuit (VC) multiplex encapsulation on CCC circuits. When you use this encapsulation type, you can configure the **ccc** family only.

atm-cisco-nlpid—Use Cisco ATM network layer protocol identifier (NLPID) encapsulation. When you use this encapsulation type, you can configure the **inet** family only.

atm-mlppp-llc—For ATM2 IQ interfaces only, use Multilink Point-to-Point (MLPPP) over AAL5 LLC. For this encapsulation type, your router must be equipped with a Link Services or Voice Services PIC. MLPPP over ATM encapsulation is not supported on ATM2 IQ OC48 interfaces.

atm-nlpid—Use ATM NLPID encapsulation. When you use this encapsulation type, you can configure the **inet** family only.

atm-ppp-llc—(ATM2 IQ interfaces and MX Series routers with MPC/MIC interfaces using the ATM MIC with SFP only) Use PPP over AAL5 LLC encapsulation.

atm-ppp-vc-mux—(ATM2 IQ interfaces and MX Series routers with MPC/MIC interfaces using the ATM MIC with SFP only) Use PPP over ATM AAL5 multiplex encapsulation.

atm-snap—(All interfaces including MX Series routers with MPC/MIC interfaces using the ATM MIC with SFP) Use ATM subnetwork attachment point (SNAP) encapsulation.

atm-tcc-snap—Use ATM SNAP encapsulation on translational cross-connect (TCC) circuits.

atm-tcc-vc-mux—Use ATM VC multiplex encapsulation on TCC circuits. When you use this encapsulation type, you can configure the **tcc** family only.

atm-vc-mux—(All interfaces including MX Series routers with MPC/MIC interfaces using the ATM MIC with SFP) Use ATM VC multiplex encapsulation. When you use this encapsulation type, you can configure the **inet** family only.

ether-over-atm-llc—(All IP interfaces including MX Series routers with MPC/MIC interfaces using the ATM MIC with SFP) For interfaces that carry IP traffic, use Ethernet over ATM LLC encapsulation. When you use this encapsulation type, you cannot configure multipoint interfaces.

ether-vpls-over-atm-llc—For ATM2 IQ interfaces only, use the Ethernet virtual private LAN service (VPLS) over ATM LLC encapsulation to bridge Ethernet interfaces and ATM interfaces over a VPLS routing instance (as described in RFC 2684, *Multiprotocol Encapsulation over ATM Adaptation Layer 5*). Packets from the ATM interfaces are converted to standard ENET2/802.3 encapsulated Ethernet frames with the frame check sequence (FCS) field removed.

ether-vpls-over-fr—For E1, T1, E3, T3, and SONET interfaces only, use the Ethernet virtual private LAN service (VPLS) over Frame Relay encapsulation to support Bridged Ethernet over Frame Relay encapsulated TDM interfaces for VPLS applications, per RFC 2427, *Multiprotocol Interconnect over Frame Relay*.

NOTE: **The SONET/SDH OC3/STM1 (Multi-Rate) MIC with SFP, the Channelized SONET/SDH OC3/STM1 (Multi-Rate) MIC with SFP, and the DS3/E3 MIC do not support Ethernet over Frame Relay encapsulation.**

ether-vpls-over-ppp—For E1, T1, E3, T3, and SONET interfaces only, use the Ethernet virtual private LAN service (VPLS) over Point-to-Point Protocol (PPP) encapsulation to support Bridged Ethernet over PPP-encapsulated TDM interfaces for VPLS applications.

ethernet—Use Ethernet II encapsulation (as described in RFC 894, *A Standard for the Transmission of IP Datagrams over Ethernet Networks*).

ethernet-ccc—Use Ethernet CCC encapsulation on Ethernet interfaces.

ethernet-vpls—Use Ethernet VPLS encapsulation on Ethernet interfaces that have VPLS enabled and that must accept packets carrying standard Tag Protocol ID (TPID) values.

NOTE: The built-in Gigabit Ethernet PIC on an M7i router does not support extended VLAN VPLS encapsulation.

ethernet-vpls-fr—Use in a VPLS setup when a CE device is connected to a PE router over a time-division multiplexing (TDM) link. This encapsulation type enables the PE router to terminate the outer layer 2 Frame Relay connection, use the 802.1p bits inside the inner Ethernet header to classify the packets, look at the MAC address from the Ethernet header, and use the MAC address to forward the packet into a given VPLS instance.

frame-relay-ccc—Use Frame Relay encapsulation on CCC circuits. When you use this encapsulation type, you can configure the **ccc** family only.

frame-relay-ether-type—Use Frame Relay ether type encapsulation for compatibility with Cisco Frame Relay. The physical interface must be configured with flexible-frame-relay encapsulation.

frame-relay-ether-type-tcc—Use Frame Relay ether type TCC for Cisco-compatible Frame Relay on TCC circuits to connect different media. The physical interface must be configured with flexible-frame-relay encapsulation.

frame-relay-ppp—Use PPP over Frame Relay circuits. When you use this encapsulation type, you can configure the **ppp** family only.

frame-relay-tcc—Use Frame Relay encapsulation on TCC circuits for connecting different media. When you use this encapsulation type, you can configure the **tcc** family only.

gre-fragmentation—For adaptive services interfaces only, use GRE fragmentation encapsulation to enable fragmentation of IPv4 packets in GRE tunnels. This encapsulation clears the do not fragment (DF) bit in the packet header. If the packet's size exceeds the tunnel's maximum transmission unit (MTU) value, the packet is fragmented before encapsulation.

multilink-frame-relay-end-to-end—Use MLFR FRF.15 encapsulation. This encapsulation is used only on multilink, link services, and voice services interfaces and their constituent T1 or E1 interfaces, and is supported on LSQ and redundant LSQ interfaces.

multilink-ppp—Use MLPPP encapsulation. This encapsulation is used only on multilink, link services, and voice services interfaces and their constituent T1 or E1 interfaces.

ppp-over-ether—Use PPP over Ethernet encapsulation to configure an underlying Ethernet interface for a dynamic PPPoE logical interface on M120 and M320 routers with Intelligent Queuing 2 (IQ2) PICs, and on MX Series routers with MPCs.

ppp-over-ether-over-atm-llc—(MX Series routers with MPCs using the ATM MIC with SFP only) For underlying ATM interfaces, use PPP over Ethernet over ATM LLC encapsulation. When you use this encapsulation type, you cannot configure the interface address. Instead, configure the interface address on the PPP interface.

vlan-bridge—Use Ethernet VLAN bridge encapsulation on Ethernet interfaces that have IEEE 802.1Q tagging, flexible-ethernet-services, and bridging enabled and that must accept packets carrying TPID 0x8100 or a user-defined TPID.

vlan-ccc—Use Ethernet virtual LAN (VLAN) encapsulation on CCC circuits. When you use this encapsulation type, you can configure the **ccc** family only.

vlan-vci-ccc—Use ATM-to-Ethernet interworking encapsulation on CCC circuits. When you use this encapsulation type, you can configure the **ccc** family only.

vlan-tcc—Use Ethernet VLAN encapsulation on TCC circuits. When you use this encapsulation type, you can configure the **tcc** family only.

vlan-vpls—Use Ethernet VLAN encapsulation on VPLS circuits.

vxlan—Use VXLAN data plane encapsulation for EVPN.

Required Privilege Level

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

Related Documentation

- *Configuring Layer 2 Switching Cross-Connects Using CCC*
- *Configuring the Encapsulation for Layer 2 Switching TCCs*
- *Configuring Interface Encapsulation on Logical Interfaces*
- *Configuring MPLS LSP Tunnel Cross-Connects Using CCC*
- *Circuit and Translational Cross-Connects Overview*
- *Identifying the Access Concentrator*
- *Configuring ATM Interface Encapsulation*
- *Configuring VLAN Encapsulation*
- *Configuring Extended VLAN Encapsulation*
- *Configuring ATM-to-Ethernet Interworking*
- *Configuring Interface Encapsulation on PTX Series Packet Transport Routers*
- *Configuring CCC Encapsulation for Layer 2 VPNs*
- *Configuring TCC Encapsulation for Layer 2 VPNs and Layer 2 Circuits*
- Configuring ATM for Subscriber Access on page 358
- *CoS on ATM IMA Pseudowire Interfaces Overview*
- *Configuring Policing on an ATM IMA Pseudowire*

enhanced-mode

Syntax enhanced-mode;

Hierarchy Level [edit firewall filter *filter-name*],
[edit firewall family *family-name* filter *filter-name*],
[edit logical-systems *logical-system-name* firewall filter *filter-name*],
[edit logical-systems *logical-system-name* firewall family *family-name* filter *filter-name*]

Release Information Statement introduced in Junos OS Release 11.4.
Statement introduced in Junos OS Release 12.3R2 for EX Series switches.

Description Limit static service filters or API-client filters to term-based filter format only for inet or inet6 families when enhanced network services mode is configured at the **[edit chassis network-services]** hierarchy level. When used with one of the chassis enhanced network services modes, firewall filters are generated in term-based format for use with MPC modules.

If enhanced network services are not configured for the chassis, the **enhanced-mode** statement is ignored and any enhanced mode firewall filters are generated in both term-based and, the default, compiled format. Only term-based (enhanced) firewall filters will be generated, regardless of the setting of the **enhanced-mode** statement at the [**edit chassis network-services**] hierarchy level, if any of the following are true:

- Flexible filter match conditions are configured at the **[edit firewall family *family-name* filter *filter-name* term *term-name* from]** or **[edit firewall filter *filter-name* term *term-name* from]** hierarchy levels.
- A tunnel header push or pop action, such as GRE encapsulate or decapsulate is configured at the **[edit firewall family *family-name* filter *filter-name* term *term-name* then]** hierarchy level.
- Payload-protocol match conditions are configured at the **[edit firewall family *family-name* filter *filter-name* term *term-name* from]** or **[edit firewall filter *filter-name* term *term-name* from]** hierarchy levels.
- An extension-header match is configured at the **[edit firewall family *family-name* filter *filter-name* term *term-name* from]** or **[edit firewall filter *filter-name* term *term-name* from]** hierarchy levels.
- A match condition is configured that only works with MPC cards, such as firewall bridge filters for IPv6 traffic.

NOTE: You cannot attach enhanced mode filters to local loopback, management, or MS-DPC interfaces. These interfaces are processed by the Routing Engine and DPC modules and can accept only compiled firewall filter format. In cases where both filter formats are needed for dynamic service filters, you can use the *enhanced-mode-override* statement on the specific filter definition to override the default filter term-based only format of chassis network-service enhanced IP mode.

NOTE: Do not use enhanced mode for firewall filters that are intended for control plane traffic. Control plane filtering is handled by the Routing Engine kernel, which cannot use the term-based format of the enhanced mode filters.

For packets sourced from the Routing Engine, the Routing Engine processes Layer 3 packets by applying output filters to the packets and forwards Layer 2 packets to the Packet Forwarding Engine for transmission. By configuring the enhanced mode filter, you explicitly specify that only the term-based filter format is used, which also implies that the Routing Engine cannot use this filter.

NOTE: The **enhanced-mode** and the **enhanced-mode-override** statements are mutually exclusive; you can define the filter with either **enhanced-mode** or **enhanced-mode-override**, but not both.

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

Related Documentation

- *enhanced-mode-override*
- *Network Services Mode Overview*
- *Firewall Filters and Enhanced Network Services Mode Overview*
- *Configuring a Filter for Use with Enhanced Network Services Mode*

family

Syntax

```
family family {
  accounting {
    destination-class-usage;
    source-class-usage {
      (input | output | input output);
    }
  }
  access-concentrator name;
  address address {
    ... the address subhierarchy appears after the main [edit interfaces interface-name unit
      logical-unit-number family family-name] hierarchy ...
  }
  bundle interface-name;
  core-facing;
  demux-destination {
    destination-prefix;
  }
  demux-source {
    source-prefix;
  }
  direct-connect;
  duplicate-protection;
  dynamic-profile profile-name;
  filter {
    group filter-group-number;
    input filter-name;
    input-list [ filter-names ];
    output filter-name;
    output-list [ filter-names ];
  }
  interface-mode (access | trunk);
  ipsec-sa sa-name;
  keep-address-and-control;
  mac-validate (loose | strict);
  max-sessions number;
  max-sessions-vsa-ignore;
  mtu bytes;
  multicast-only;
  negotiate-address;
  no-redirects;
  policer {
    arp policer-template-name;
    input policer-template-name;
    output policer-template-name;
  }
  primary;
  protocols [inet iso mpls];
  proxy inet-address address;
  receive-options-packets;
  receive-ttl-exceeded;
  remote (inet-address address | mac-address address);
  rpf-check {
```

```
    fail-filter filter-name
    mode loose;
  }
  sampling {
    input;
    output;
  }
  service {
    input {
      post-service-filter filter-name;
      service-set service-set-name <service-filter filter-name>;
    }
    output {
      service-set service-set-name <service-filter filter-name>;
    }
  }
  service-name-table table-name
  short-cycle-protection <lockout-time-min minimum-seconds lockout-time-max
    maximum-seconds>;
  (translate-discard-eligible | no-translate-discard-eligible);
  (translate-fecn-and-becn | no-translate-fecn-and-becn);
  translate-plp-control-word-de;
  unnumbered-address interface-name destination address destination-profile profile-name;
  vlan-id number;
  vlan-id-list [number number-number];
  address address {
    arp ip-address (mac | multicast-mac) mac-address <publish>;
    broadcast address;
    destination address;
    destination-profile name;
    eui-64;
    master-only;
    multipoint-destination address dlci dlci-identifier;
    multipoint-destination address {
      epd-threshold cells;
      inverse-arp;
      oam-liveness {
        up-count cells;
        down-count cells;
      }
      oam-period (disable | seconds);
      shaping {
        (cbr rate | rtvbr burst length peak rate sustained rate | vbr burst length peak rate
          sustained rate);
        queue-length number;
      }
      vci vpi-identifier.vci-identifier;
    }
    preferred;
    primary;
    vrrp-groupgroup-id {
      (accept-data | no-accept-data);
      advertise-interval seconds;
      authentication-key key;
      authentication-type authentication;
      fast-interval milliseconds;
```



```
        (preempt | no-preempt) {
          hold-time seconds;
        }
        priority number;
        track {
          interface interface-name {
            bandwidth-threshold bits-per-second priority-cost priority;
            priority-cost priority;
          }
          priority-hold-time seconds;
          route prefix routing-instance instance-name priority-cost priority;
          }
        }
        virtual-address [ addresses ];
        }
      virtual-link-local-address ipv6-address;
    }
  }
```

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

Release Information Statement introduced before Junos OS Release 7.4.
Option **max-sessions-vsa-ignore** introduced in Junos OS Release 11.4.

Description Configure protocol family information for the logical interface.

NOTE: Not all subordinate stanzas are available to every protocol family.

Options *family*—Protocol family:

- **any**—Protocol-independent family used for Layer 2 packet filtering

NOTE: This option is not supported on T4000 Type 5 FPCs.

- **bridge**—(M Series and T Series routers only) Configure only when the physical interface is configured with **ethernet-bridge** type encapsulation or when the logical interface is configured with **vlan-bridge** type encapsulation
- **ethernet-switching**—(M Series and T Series routers only) Configure only when the physical interface is configured with **ethernet-bridge** type encapsulation or when the logical interface is configured with **vlan-bridge** type encapsulation
- **ccc**—Circuit cross-connect protocol suite
- **inet**—Internet Protocol version 4 suite
- **inet6**—Internet Protocol version 6 suite
- **iso**—International Organization for Standardization Open Systems Interconnection (ISO OSI) protocol suite
- **mlfr-end-to-end**—Multilink Frame Relay FRF.15
- **mlfr-uni-nni**—Multilink Frame Relay FRF.16
- **multilink-ppp**—Multilink Point-to-Point Protocol
- **mpls**—Multiprotocol Label Switching (MPLS)
- **pppoe**—Point-to-Point Protocol over Ethernet
- **tcc**—Translational cross-connect protocol suite
- **tnp**—Trivial Network Protocol
- **vpls**—(M Series and T Series routers only) Virtual private LAN service

The remaining statements are explained separately.

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

Related Documentation
- *Configuring the Protocol Family*

family (Dynamic Demux Interface)

Syntax

```
family family {
  access-concentrator name;
  address address;
  demux-source {
    source-address;
  }
  direct-connect;
  duplicate-protection;
  dynamic-profile profile-name;
  filter {
    input filter-name;
    output filter-name;
  }
  mac-validate (loose | strict);
  max-sessions number;
  max-sessions-vsa-ignore;
  service-name-table table-name;
  short-cycle-protection <lockout-time-min minimum-seconds lockout-time-max
    maximum-seconds>;
  unnumbered-address interface-name <preferred-source-address address>;
}
```

Hierarchy Level [edit dynamic-profiles *profile-name* interfaces demux0 unit *logical-unit-number*]

Release Information Statement introduced in Junos OS Release 9.3.
Option **pppoe** introduced in Junos OS Release 11.2.

Description Configure protocol family information for the logical interface.

NOTE: Not all subordinate stanzas are available to every protocol family.

Options *family*—Protocol family:

- **inet**—Internet Protocol version 4 suite
- **inet6**—Internet Protocol version 6 suite
- **pppoe**—(MX Series routers with MPCs only) Point-to-Point Protocol over Ethernet

The remaining statements are explained separately.

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

Related Documentation

- Configuring Dynamic Subscriber Interfaces Using IP Demux Interfaces in Dynamic Profiles on page 77
- Subscriber Interfaces and Demultiplexing Overview on page 70

family (Dynamic PPPoE)

Syntax

```
family family {
  unnumbered-address interface-name;
  address address;
  service {
    input {
      service-set service-set-name {
        service-filter filter-name;
      }
      post-service-filter filter-name;
    }
    output {
      service-set service-set-name {
        service-filter filter-name;
      }
    }
  }
  filter {
    input filter-name {
      precedence precedence;
    }
    output filter-name {
      precedence precedence;
    }
  }
}
```

Hierarchy Level [edit dynamic-profiles *profile-name* interfaces pp0 unit "$junos-interface-unit"]

Release Information Statement introduced in Junos OS Release 10.1.

Description Configure protocol family information for the logical interface.

Options *family*—Protocol family:

- **inet**—Internet Protocol version 4 suite
- **inet6**—Internet Protocol version 6 suite

The remaining statements are explained separately.

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

Related Documentation

- Configuring a Basic PPPoE Dynamic Profile on page 146
- Configuring a PPPoE Dynamic Profile with Additional Options on page 148
- Dynamic PPPoE Subscriber Interfaces over Static Underlying Interfaces Overview on page 153

family (Dynamic Standard Interface)

Syntax

```
family family {
  access-concentrator name;
  address address;
  direct-connect;
  duplicate-protection;
  dynamic-profile profile-name;
  filter {
    adf {
      counter;
      input-precedence precedence;
      not-mandatory;
      output-precedence precedence;
      rule rule-value;
    }
    input filter-name {
      precedence precedence;
    }
    output filter-name {
      precedence precedence;
    }
  }
  mac-validate (loose | strict);
  max-sessions number;
  max-sessions-vsa-ignore;
  rpf-check {
    fail-filter filter-name;
    mode loose;
  }
  service {
    input {
      service-set service-set-name {
        service-filter filter-name;
      }
      post-service-filter filter-name;
    }
    output {
      service-set service-set-name {
        service-filter filter-name;
      }
    }
  }
  service-name-table table-name
  short-cycle-protection <lockout-time-min minimum-seconds lockout-time-max
    maximum-seconds>;
  unnumbered-address interface-name <preferred-source-address address>;
}
```

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

Release Information Statement introduced in Junos OS Release 9.2.
Option **pppoe** introduced in Junos OS Release 11.2.

Description Configure protocol family information for the logical interface.

NOTE: **Not all subordinate stanzas are available to every protocol family.**

Options ***family***—Protocol family:

- **inet**—IP version 4 suite
- **inet6**—IP version 6 suite
- **pppoe**—(MX Series routers with MPCs only) Point-to-Point Protocol over Ethernet
- **vpls**—Virtual private LAN service

The remaining statements are explained separately.

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

Related Documentation

- *Example: Configuring Static Routing on Logical Systems*
- *Configuring the Protocol Family*

filter (Applying to a Logical Interface)

Syntax

```
filter {
  group filter-group-number;
  input filter-name;
  input-list [ filter-names ];
  output filter-name;
  output-list [ filter-names ];
}
```

Hierarchy Level

Protocol-independent firewall filter on MX Series router logical interface:

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

All other standard firewall filters on all other devices:

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

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 12.3R2 for EX Series switches.

Description

Apply a stateless firewall filter to a logical interface at a specific protocol level.

Options

group *filter-group-number*—Number of the group to which the interface belongs. Range: 1 through 255

input *filter-name*—Name of one filter to evaluate when packets are received on the interface.

input-list [*filter-names*]—Names of filters to evaluate when packets are received on the interface. Up to 16 filters can be included in a filter input list.

output *filter-name*—Name of one filter to evaluate when packets are transmitted on the interface.

output-list [*filter-names*]—Names of filters to evaluate when packets are transmitted on the interface. Up to 16 filters can be included in a filter output list.

Required Privilege Level

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

Related Documentation

- *Guidelines for Configuring Firewall Filters*
- *Guidelines for Applying Firewall Filters*

filter (Dynamic Firewalls)

Syntax

```
filter {
  adf {
    counter;
    input-precedence precedence;
    not-mandatory;
    output-precedence precedence;
    rule rule-value;
  }
  input filter-name {
    precedence precedence;
    shared-name filter-shared-name;
  }
  output filter-name {
    precedence precedence;
    shared-name filter-shared-name;
  }
}
```

Hierarchy Level

[edit dynamic-profiles *profile-name* interfaces *interface-name* unit *logical-unit-number* family *family*],
[edit dynamic-profiles *profile-name* interfaces demux0 unit *logical-unit-number* family *family*],
[edit dynamic-profiles *profile-name* interfaces pp0 unit “$junos–interface–unit” family *family*]

Release Information

Statement introduced in Junos OS Release 9.2.
Support at the **[edit dynamic-profiles *profile-name* interfaces pp0 unit “$junos-interface-unit” family *family*]** hierarchy level introduced in Junos OS Release 10.1.
shared-name statement added in Junos OS Release 12.2.

Description

Apply a dynamic filter to an interface. You can configure filters for either **family inet** or **family inet6**, and the filters can be classic filters, fast update filters, or (for the **adf** statement) Ascend-Data-Filters. Only the Internet Protocol version 4 (IPv4) protocol family is currently supported for dynamic PPPoE logical interfaces.

Options

input *filter-name*—Name of one filter to evaluate when packets are received on the interface.

output *filter-name*—Name of one filter to evaluate when packets are transmitted on the interface.

The remaining statements are explained separately.

Required Privilege Level

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

Related Documentation

- *Firewall Filters Overview*
- *Understanding Dynamic Firewall Filters*
- *Classic Filters Overview*

- *Basic Classic Filter Syntax*

flexible-vlan-tagging

Syntax flexible-vlan-tagging;

Hierarchy Level [edit interfaces ae*x*],
[edit interfaces ge-*fpc/pic/port*],
[edit interfaces et-*fpc/pic/port*],
[edit interfaces ps0],
[edit interfaces xe-*fpc/pic/port*]

Release Information Statement introduced in Junos OS Release 8.1.
Support for aggregated Ethernet added in Junos OS Release 9.0.
Statement introduced in Junos OS Release 12.1x48 for PTX Series Packet Transport Routers.
Statement introduced in Junos OS Release 13.2X50-D15 for EX Series switches.
Statement introduced in Junos OS Release 13.2X51-D20 for the QFX Series.

Description Support simultaneous transmission of 802.1Q VLAN single-tag and dual-tag frames on logical interfaces on the same Ethernet port, and on pseudowire logical interfaces.

This statement is supported on M Series and T Series routers, for Fast Ethernet and Gigabit Ethernet interfaces only on Gigabit Ethernet IQ2 and IQ2-E, IQ, and IQE PICs, and for aggregated Ethernet interfaces with member links in IQ2, IQ2-E, and IQ PICs or in MX Series DPCs, or on Ethernet interfaces for PTX Series Packet Transport Routers or 100-Gigabit Ethernet Type 5 PIC with CFP. This statement is supported on Gigabit Ethernet, 10-Gigabit Ethernet, 40-Gigabit Ethernet, and aggregated Ethernet interfaces on EX Series switches.

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

Related Documentation

- *Configuring Mixed Tagging*
- *Configuring Flexible VLAN Tagging on PTX Series Packet Transport Routers*

forwarding-classes (Class-of-Service)

Syntax

```
forwarding-classes {
  class queue-num queue-number priority (high | low);
  queue queue-number class-name priority (high | low) [ policing-priority (premium | normal) ];
}
```

Hierarchy Level [edit class-of-service]

Release Information Statement introduced before Junos OS Release 7.4.
policing-priority option introduced in Junos OS Release 9.5.
Statement introduced on PTX Series Packet Transport Routers in Junos OS Release 12.1.

Description Associate the forwarding class with a queue name and number. For M320, MX Series, T Series routers and EX Series switches only, you can configure fabric priority queuing by including the **priority** statement. For Enhanced IQ PICs, you can include the **policing-priority** option.

NOTE: **The priority add policing-priority options are not supported on PTX Series Packet Transport Routers.**

The statements are explained separately.

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

Related Documentation

- *Configuring Forwarding Classes*
- *Forwarding Classes and Fabric Priority Queues*
- *Classifying Packets by Egress Interface*

fragmentation-maps

Syntax

```
fragmentation-maps {
  map-name {
    forwarding-class class-name {
      (fragment-threshold bytes | no-fragmentation);
      multilink-class number;
    }
  }
}
```

Hierarchy Level [edit class-of-service]

Release Information Statement introduced before Junos OS Release 7.4.

Description For link services IQ (**lsq**) interfaces only, define fragmentation properties for individual forwarding classes.

Default If you do not include this statement, traffic in all forwarding classes is fragmented.

Options ***map-name***—Name of the fragmentation map.

The remaining statements are explained separately.

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

Related Documentation

- *Configuring CoS Fragmentation by Forwarding Class on LSQ Interfaces*

group (DHCP Local Server)

Syntax

```
group group-name {
  authentication {
    password password-string;
    username-include {
      circuit-type;
      client-id;
      delimiter delimiter-character;
      domain-name domain-name-string;
      logical-system-name;
      mac-address;
      option-60;
      option-82 <circuit-id> <remote-id>;
      relay-agent-interface-id
      relay-agent-remote-id;
      relay-agent-subscriber-id;
      routing-instance-name;
      user-prefix user-prefix-string;
    }
  }
  dynamic-profile profile-name <aggregate-clients (merge | replace) | use-primary
    primary-profile-name>;
  interface interface-name {
    exclude;
    overrides {
      client-discover-match (option60-and-option82 | incoming-interface);
      interface-client-limit number;
      process-inform {
        pool pool-name;
      }
      rapid-commit;
    }
    service-profile dynamic-profile-name;
    trace;
    upto upto-interface-name;
  }
  liveness-detection {
    failure-action (clear-binding | clear-binding-if-interface-up | log-only);
    method {
      bfd {
        version (0 | 1 | automatic);
        minimum-interval milliseconds;
        minimum-receive-interval milliseconds;
        multiplier number;
        no-adaptation;
        transmit-interval {
          minimum-interval milliseconds;
          threshold milliseconds;
        }
        detection-time {
          threshold milliseconds;
        }
        session-mode(automatic | multihop | singlehop);
```

```
        holddown-interval milliseconds;
      }
    }
  }
  overrides {
    client-discover-match (option60-and-option82 | incoming-interface);
    delegated-pool;
    interface-client-limit number;
    process-inform {
      pool pool-name;
    }
    rapid-commit;
  }
  reconfigure {
    attempts attempt-count;
    clear-on-abort;
    strict;
    timeout timeout-value;
    token token-value;
    trigger {
      radius-disconnect;
    }
  }
  route-suppression;
  service-profile dynamic-profile-name;
}
```

Hierarchy Level

[edit system services dhcp-local-server],
[edit system services dhcp-local-server dhcpv6],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* system services dhcp-local-server ...],
[edit logical-systems *logical-system-name* system services dhcp-local-server ...],
[edit routing-instances *routing-instance-name* system services dhcp-local-server ...]

Release Information

Statement introduced in Junos OS Release 9.0.
Statement introduced in Junos OS Release 12.1 for EX Series switches.

Description

Configure a group of interfaces that have a common configuration, such as authentication parameters. A group must contain at least one interface.

Options

group-name—Name of the group.

The remaining statements are explained separately.

Required Privilege Level

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

Related Documentation

- *Extended DHCP Local Server Overview*
- *Grouping Interfaces with Common DHCP Configurations*
- *Using External AAA Authentication Services with DHCP*
- Attaching Dynamic Profiles to DHCP Subscriber Interfaces or DHCP Client Interfaces on page 114
- *Configuring a DHCP Server on Switches (CLI Procedure)*

hierarchical-scheduler (Subscriber Interfaces on MX Series Routers)

Syntax

```
hierarchical-scheduler {
  implicit-hierarchy;
  maximum–hierarchy–levels number;
}
```

Hierarchy Level [edit interfaces *interface-name*]

Release Information Statement introduced in Junos OS Release 10.1.
Option **implicit-hierarchy** introduced in Junos OS Release 13.1.
Support on GRE tunnel interfaces configured on physical interfaces on MICs or MPCs in MX Series routers added in Junos OS Release 13.3.

Description Configure hierarchical scheduling options on the interface.

The statement is supported on the following interfaces:

- MIC and MPC interfaces in MX Series routers
- GRE tunnel interfaces configured on physical interfaces hosted on MIC or MPC line cards in MX Series routers

To enable hierarchical scheduling on MX Series routers, configure the **hierarchical-scheduler** statement at each member physical interface level of a particular aggregated Ethernet interface as well as at that aggregated Ethernet interface level. On other routing platforms, it is enough if you include this statement at the aggregated Ethernet interface level.

Options **implicit-hierarchy**—Configure three-level hierarchical scheduling. When you include the **implicit-hierarchy** option, a hierarchical relationship is formed between the CoS scheduler nodes at level 1, level 2, and level 3. The **implicit-hierarchy** option is supported only on MPC/MIC subscriber interfaces and interface sets running over aggregated Ethernet on MX Series routers.

maximum-hierarchy-levels *number*—Configure two-level hierarchical scheduling. Specify the maximum number of hierarchical scheduling levels allowed for node scaling. The only supported value is 2. The **maximum-hierarchy-levels** option is supported on MPC/MIC or EQ DPC subscriber interfaces and interface sets running over aggregated Ethernet on MX Series routers.

- If you include the **maximum-hierarchy-levels** option, interface sets are allowed only at level 3; they are not allowed at level 2. In this case, if you configure a level 2 interface set, you generate Packet Forwarding Engine errors.
- If you do not include the **maximum-hierarchy-levels** option, interface sets can be at either level 2 or level 3, depending on whether the member logical interfaces within the interface set have a traffic control profile. If any member logical interface has a traffic control profile, then the interface set is a level 2 CoS scheduler node. If no member logical interface has a traffic control profile, the interface set is at level 3.

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

Related Documentation

- *Understanding Two-Level and Three-Level Hierarchical CoS for Subscriber Interfaces*
- *Configuring Hierarchical CoS for a Subscriber Interface of Aggregated Ethernet Links*
- *Configuring Hierarchical Schedulers for CoS*
- *Configuring Hierarchical CoS on a Static PPPoE Subscriber Interface*
- Hierarchical CoS on MPLS Pseudowire Subscriber Interfaces Overview on page 333

inline-services (PIC level)

Syntax

```
inline-services {
  bandwidth (1g | 10g);
}
```

Hierarchy Level [edit chassis fpc *slot-number* pic *number*]

Release Information Statement introduced in Junos OS Release 11.4.

Description Enable inline services on PICs residing on MPCs. To enable inline services that are specified at the fpc level, see configuration statement *inline-services (FPC Level)*

The remaining statement is explained separately.

Options The option is described separately.

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

Related Documentation

- *Enabling Inline Service Interfaces*
- *Configuring an L2TP LNS with Inline Service Interfaces*

inner-tag-protocol-id (Dynamic VLANs)

Syntax inner-tag-protocol-id *tpids*;

Hierarchy Level [edit dynamic-profiles *profile-name* interfaces *interface-name* unit *logical-unit-number*input-vlan-map],
[edit dynamic-profiles *profile-name* interfaces *interface-name* unit *logical-unit-number*output-vlan-map]

Release Information Statement introduced in Junos OS Release 10.4.

Description For dynamic VLAN interfaces, configure the IEEE 802.1Q TPID value to rewrite for the inner tag. All TPIDs you include in input and output VLAN maps must be among those you specify at the **[edit interfaces *interface-name* gigether-options ethernet-switch-profile tag-protocol-id *tpids*]** hierarchy level.

Default If the **inner-tag-protocol-id** statement is not configured, the TPID value is 0x8100.

Options ***tpids***—TPIDs to be accepted on the VLAN. Specify TPIDs in hexadecimal format.

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

Related Documentation
- *Configuring Inner and Outer TPIDs and VLAN IDs*

inner-vlan-id (Dynamic VLANs)

Syntax inner-vlan-id *number*;

Hierarchy Level [edit dynamic-profiles *profile-name* interfaces *interface-name* unit *logical-unit-number*input-vlan-map],
[edit dynamic-profiles *profile-name* interfaces *interface-name* unit *logical-unit-number*output-vlan-map]

Release Information Statement introduced in Junos OS Release 10.4.

Description For dynamic VLAN interfaces, specify the VLAN ID to rewrite for the inner tag of the final packet.

You cannot include the **inner-vlan-id** statement with the **swap** statement, **swap-push** statement, **push-push** statement, or **push-swap** statement and the **inner-vlan-id** statement at the **[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number* output-vlan-map]** hierarchy level. If you include any of those statements in the output VLAN map, the VLAN ID in the outgoing frame is rewritten to the **inner-vlan-id** statement you include at the **[edit interfaces *interface-name* unit *logical-unit-number*]** hierarchy level.

Options *number*—VLAN ID number.

Range: 0 through 4094

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

Related Documentation
- *Configuring Inner and Outer TPIDs and VLAN IDs*

input (Dynamic Service Sets)

Syntax

```
input {
  service-set service-set-name {
    service-filter filter-name;
  }
  post-service-filter filter-name;
}
```

Hierarchy Level

[edit dynamic-profiles *profile-name* interfaces *interface-name* unit *logical-unit-number* family *family* service],
[edit dynamic-profiles *profile-name* interfaces pp0 unit “$junos–interface–unit” family *family* service]

Release Information

Statement introduced in Junos OS Release 9.5.
Support at the **[edit dynamic-profiles *profile-name* interfaces pp0 unit “$junos-interface-unit” family *family* service]** hierarchy level introduced in Junos OS Release 10.1.

Description

Define the input service sets and filters to be applied to traffic by a dynamic profile. Only the Internet Protocol version 4 (IPv4) protocol family is currently supported for dynamic PPPoE logical interfaces.

The remaining statements are explained separately.

Required Privilege Level

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

Related Documentation

- *Dynamic Service Sets Overview*
- *Associating Service Sets with Interfaces in a Dynamic Profile*

input-hierarchical-policer

Syntax input-hierarchical-policer *policer-name*;

Hierarchy Level [edit interfaces *interface-name* layer2-policer],
[edit interfaces *interface-name* unit *logical-unit-number* layer2-policer],

Release Information Statement introduced in Junos OS Release 9.5.
Statement introduced in Junos OS Release 12.3R2 for EX Series switches.

Description Apply a hierarchical policer to the Layer 2 input traffic for all protocol families at the physical or logical interface.

Options ***policer-name***—Name of the hierarchical policer.

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

Related Documentation

- *Hierarchical Policers*
- *layer2-policer (Hierarchical Policer)*

input-vlan-map (Dynamic Interfaces)

Syntax

```
input-vlan-map {
  inner-tag-protocol-id tpid;
  inner-vlan-id number;
  (push | swap);
  tag-protocol-id tpid;
  vlan-id number;
}
```

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

Release Information Statement introduced in Junos OS Release 10.4.

Description For dynamic interfaces, define the rewrite profile to be applied to incoming frames on this logical interface.

The statements are explained separately.

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

Related Documentation

- *Stacking and Rewriting VLAN Tags for the Layer 2 Wholesale Solution*

interface (Dynamic Interface Sets)

Syntax

```
interface interface-name {
  unit logical unit number {
    advisory-options {
      downstream-rate rate;
      upstream-rate rate;
    }
  }
}
```

Hierarchy Level [edit dynamic-profiles *profile-name* interfaces interface-set *interface-set-name*]

Release Information Statement introduced in Junos OS Release 12.2.

Description Add a subscriber interface to a dynamic interface set.

In a dynamic profile that defines an agent circuit identifier (ACI) interface set, observe the following guidelines when you use the **interface** statement:

- Use the predefined dynamic interface variable **$junos-interface-ifd-name** to represent the interface name. Do not use a specific interface name, such as **demux0**, when defining an ACI interface set.
- Do not include the **unit *logical-unit-number*** statement.

Options *interface-name*—Either the specific name of the interface to include in the interface set, or the predefined dynamic interface variable **$junos-interface-ifd-name**. The interface variable is dynamically replaced with the interface that the DHCP or PPPoE subscriber accesses when connecting to the router.

The remaining statement is explained separately.

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

Related Documentation

- Defining Agent Circuit Identifier Interface Sets on page 31
- *Guidelines for Configuring Dynamic CoS for Subscriber Access*
- *Configuring an Interface Set of Subscribers in a Dynamic Profile*
- Agent Circuit Identifier-Based Dynamic VLANs Components Overview on page 27

interface-name

Syntax interface-name;

Hierarchy Level [edit interfaces *interface-name* auto-configure vlan-ranges authentication username-include],
[edit interfaces *interface-name* auto-configure stacked-vlan-ranges authentication username-include],

Release Information Statement introduced in Junos OS Release 10.0.

Description Append the interface name and VLAN ID or stacked VLAN ID to the username string used for authentication. The appended information takes the following format:

- For single VLAN—**<interface-name>:<4-digit-vlan-id>**
- For stack VLANs—**<interface-name>:<4-digit-svlan-id>-<4-digit-vlan-id>**

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

Related Documentation
- Configuring VLAN Interface Username Information for AAA Authentication on page 22

interface-set (Dynamic VLAN Interface Sets Association)

Syntax

```
interface-set interface-set-name {
  interface interface-name {
    unit logical-unit-number {
      advisory-options {
        downstream-rate rate;
        upstream-rate rate;
      }
    }
  }
}
```

Hierarchy Level [edit dynamic-profiles *profile-name* interfaces]

Release Information Statement introduced in Junos OS Release 12.2.

Description For MX Series routers with MPC/MIC modules that face the access side of the network, associate an agent circuit identifier (ACI) interface set with a dynamic VLAN subscriber interface for DHCP or PPPoE subscribers. To associate an ACI interface set with a dynamic subscriber interface, you must include the **interface-set** stanza in the dynamic profile that defines the logical subscriber interface.

An ACI interface set is a logical collection of subscriber interfaces that originate at the same household or on the same access-loop port.

Options

- *interface-set-name*—Name of the ACI interface set, which is represented in a dynamic profile for a subscriber interface by the Junos OS predefined variable **$junos-interface-set-name**.

The remaining statements are explained separately.

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

Related Documentation

- Configuring Dynamic VLAN Subscriber Interfaces Based on Agent Circuit Identifier Information on page 36
- Agent Circuit Identifier-Based Dynamic VLANs Components Overview on page 27

interface-set (Dynamic VLAN Interface Sets Definition)

Syntax

```
interface-set interface-set-name {
  interface interface-name;
  pppoe-underlying-options {
    max-sessions number;
  }
}
```

Hierarchy Level [edit dynamic-profiles *profile-name* interfaces]

Release Information Statement introduced in Junos OS Release 12.2.

Description For MX Series routers with MPC/MIC modules that face the access side of the network, configure an agent circuit identifier (ACI) interface set for the creation of dynamic VLAN subscriber interfaces for DHCP or PPPoE subscribers based on ACI information. An ACI interface set is a logical collection of subscriber interfaces that originate at the same household or on the same access-loop port.

To configure an ACI interface set for dynamic VLAN subscriber interfaces, you must include the **interface-set** stanza in the dynamic profile that defines the ACI interface set.

Options

- *interface-set-name*—Name of the ACI interface set, which is represented in a dynamic profile by the Junos OS predefined variable **$junos-interface-set-name**.

The remaining statements are explained separately.

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

Related Documentation

- Defining Agent Circuit Identifier Interface Sets on page 31
- Clearing Agent Circuit Identifier Interface Sets on page 39
- Agent Circuit Identifier-Based Dynamic VLANs Components Overview on page 27

interfaces

Syntax interfaces { ... }

Hierarchy Level [edit]

Release Information Statement introduced before Junos OS Release 7.4.

Description Configure interfaces on the router or switch.

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

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

Related Documentation

- *Physical Interface Configuration Statements Overview*
- *Configuring Aggregated Ethernet Link Protection*

interfaces (Static and Dynamic Subscribers)

Syntax

```
interfaces {
  interface-name {
    unit logical-unit-number {
      auto-configure {
        agent-circuit-identifier {
          dynamic-profile profile-name;
        }
      }
      family family {
        access-concentrator name;
        address address;
        direct-connect;
        duplicate-protection;
        dynamic-profile profile-name;
        filter {
          adf {
            counter;
            input-precedence precedence;
            not-mandatory;
            output-precedence precedence;
            rule rule-value;
          }
          input filter-name (
            precedence precedence;
            shared-name filter-shared-name;
          }
          output filter-name {
            precedence precedence;shared-name filter-shared-name;
          }
        }
        max-sessions number;
        max-sessions-vsa-ignore;
        rpf-check {
          mode loose;
        }
        service {
          input {
            service-set service-set-name {
              service-filter filter-name;
            }
            post-service-filter filter-name;
          }
          output {
            service-set service-set-name {
              service-filter filter-name;
            }
          }
        }
        service-name-table table-name
        short-cycle-protection <lockout-time-min minimum-seconds lockout-time-max
          maximum-seconds>;
        unnumbered-address interface-name <preferred-source-address address>;
```

```
        }
        filter {
          input filter-name;
          shared-name filter-shared-name;
          output filter-name;
          shared-name filter-shared-name;
        }
        ppp-options {
          chap;
          pap;
        }
        proxy-arp;
        vlan-id;
        vlan-tags outer [tpid].vlan-id [inner [tpid].vlan-id];
      }
      vlan-tagging;
    }
    interface-set interface-set-name {
      interface interface-name {
        unit logical unit number {
          advisory-options {
            downstream-rate rate;
            upstream-rate rate;
          }
        }
      }
      pppoe-underlying-options {
        max-sessions number;
      }
    }
    demux0 {
      unit logical-unit-number {
        demux-options {
          underlying-interface interface-name
        }
        family family {
          access-concentrator name;
          address address;
          direct-connect;
          duplicate-protection;
          dynamic-profile profile-name;
          demux-source {
            source-prefix;
          }
          filter{
            input filter-name (
              precedence precedence;
              shared-name filter-shared-name;
            }
            output filter-name {
              precedence precedence;
              shared-name filter-shared-name;
            }
          }
          mac-validate (loose | strict):
          max-sessions number;
```

```
                max-sessions-vsa-ignore;
                rpf-check {
                    fail-filter filter-name;
                    mode loose;
                }
                service-name-table table-name
                short-cycle-protection <lockout-time-min minimum-seconds lockout-time-max
                    maximum-seconds>;
                unnumbered-address interface-name <preferred-source-address address>;
            }
            filter {
                input filter-name;
                output filter-name;
            }
            vlan-id number;
            vlan-tags outer [tpid].vlan-id [inner [tpid].vlan-id];
        }
    }
    pp0 {
        unit logical-unit-number {
            keepalives interval seconds;
            no-keepalives;
            pppoe-options {
                underlying-interface interface-name;
                server;
            }
            ppp-options {
                authentication [ authentication-protocols ];
                chap {
                    challenge-length minimum minimum-length maximum maximum-length;
                }
                pap;
            }
            family inet {
                unnumbered-address interface-name;
                address address;
                service {
                    input {
                        service-set service-set-name {
                            service-filter filter-name;
                        }
                        post-service-filter filter-name;
                    }
                    output {
                        service-set service-set-name {
                            service-filter filter-name;
                        }
                    }
                }
                filter {
                    input filter-name {
                        precedence precedence;
                        shared-name filter-shared-name;
                    }
                    output filter-name {
                        precedence precedence;
```



```
                    shared-name filter-shared-name;
                  }
                }
              }
            }
          }
        }
```

Hierarchy Level [edit dynamic-profiles *profile-name*]

Release Information Statement introduced in Junos OS Release 9.2.

Description Define interfaces for dynamic profiles.

Options *interface-name*—The interface variable (**$junos-interface-ifd-name**). The interface variable is dynamically replaced with the interface the DHCP client accesses when connecting to the router.

NOTE: **Though we do not recommend it, you can also enter the specific name of the interface you want to assign to the dynamic profile.**

The remaining statements are explained separately.

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

Related Documentation

- *Configuring Static Subscriber Interfaces in Dynamic Profiles*
- Configuring Dynamic Subscriber Interfaces Using IP Demux Interfaces in Dynamic Profiles on page 77
- Configuring Dynamic PPPoE Subscriber Interfaces Using Dynamic Profiles on page 145
- Configuring Dynamic VLANs Based on Agent Circuit Identifier Information on page 29
- DHCP Subscriber Interface Overview on page 69
- *Relationship Between Subscribers and Interfaces in an Access Network*
- *Configuring Subscribers over Static Interfaces*
- *Demultiplexing Interface Overview*

keepalives

Syntax keepalives <interval *seconds*> <down-count *number*> <up-count *number*>;

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

Release Information Statement introduced before Junos OS Release 7.4.

Description Enable the sending of keepalives on a physical interface configured with PPP, Frame Relay, or Cisco HDLC encapsulation.

For ATM2 IQ interfaces only, you can enable keepalives on a logical interface unit if the logical interface is configured with one of the following PPP over ATM encapsulation types:

- **atm-ppp-llc**—PPP over AAL5 LLC encapsulation.
- **atm-ppp-vc-mux**—PPP over AAL5 multiplex encapsulation.

Default Sending of keepalives is enabled by default. The default keepalive interval is 10 seconds for PPP, Frame Relay, or Cisco HDLC. The default down-count is 3 and the default up-count is 1 for PPP or Cisco HDLC.

Options **down-count *number***—The number of keepalive packets a destination must fail to receive before the network takes down a link.

Range: 1 through 255

Default: 3

interval *seconds*—The time in seconds between successive keepalive requests.

Range: 1 through 32767 seconds

Default: 10 seconds

up-count *number*—The number of keepalive packets a destination must receive to change a link's status from down to up.

Range: 1 through 255

Default: 1

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

Related Documentation

- *Configuring Keepalives*
- *Configuring Frame Relay Keepalives*
- *Applying PPP Attributes to L2TP LNS Subscribers Per Inline Service Interface*

keepalives (Dynamic Profiles)

Syntax

```
keepalives {
  interval seconds;
}
```

Hierarchy Level

[edit dynamic-profiles *profile-name* interfaces pp0 unit *logical-unit-number*]
[edit dynamic-profiles *profile-name* interfaces pp0 unit “$junos-interface-unit”]
[edit dynamic-profiles *profile-name* interfaces "$junos-interface-ifd-name" unit “$junos-interface-unit”]

Release Information

Statement introduced in Junos OS Release 9.5.
Support at the **[edit dynamic-profiles *profile-name* interfaces pp0 unit “$junos-interface-unit”]** hierarchy level introduced in Junos OS Release 10.1.
Support at the **[edit dynamic-profiles *profile-name* interfaces "$junos-interface-ifd-name" unit “$junos-interface-unit”]** hierarchy level introduced in Junos OS Release 12.2.

Description

Specify the keepalive interval in a PPP dynamic profile.

Default

Sending of keepalives is enabled by default.

Options

interval *seconds*—The time in seconds between successive keepalive requests.

Range: 1 through 32767 seconds

Default: 30 seconds for LNS-based PPP sessions. 10 seconds for all other PPP sessions.

Required Privilege Level

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

Related Documentation

- *Dynamic Profiles Overview*
- *Configuring Dynamic Authentication for PPP Subscribers*
- *Applying PPP Attributes to L2TP LNS Subscribers Per Inline Service Interface*

local-name

Syntax

local-name *name*;

Hierarchy Level

[edit interfaces *interface-name* ppp-options chap],
[edit interfaces *interface-name* ppp-options pap],
[edit interfaces *interface-name* unit *logical-unit-number* ppp-options chap],
[edit interfaces *interface-name* unit *logical-unit-number* ppp-options pap],
[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number* ppp-options chap],
[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number* ppp-options pap]

Release Information

Statement introduced before Junos OS Release 7.4.
Support for PAP added in Junos OS Release 8.3.

Description

For CHAP authentication, the value sent in CHAP challenge and response packets on a per interface basis. For PAP authentication, the local hostname for sending PAP authentication requests.

For ATM2 IQ interfaces only, you can configure a CHAP local name on the logical interface unit if the logical interface is configured with one of the following PPP over ATM encapsulation types:

- **atm-ppp-llc**—PPP over AAL5 LLC encapsulation.
- **atm-ppp-vc-mux**—PPP over AAL5 multiplex encapsulation.

Default

For CHAP authentication, if you do not include the **local-name** statement in the configuration, the interface sends the router's system hostname in CHAP challenge and response packets.

Required Privilege Level

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

Related Documentation

- *Configuring the PPP Challenge Handshake Authentication Protocol*
- *Configuring the PPP Password Authentication Protocol*
- *Junos OS Administration Library for Routing Devices*

mac

Syntax mac *mac-address*;

Hierarchy Level [edit interfaces *interface-name*]

Release Information Statement introduced before Junos OS Release 7.4.

Description Set the MAC address of the interface.

Use this statement at the **[edit interfaces ... ps0]** hierarchy level to configure the MAC address for a pseudowire logical device that is used for subscriber interfaces over point-to-point MPLS pseudowires.

Options *mac-address*—MAC address. Specify the MAC address as six hexadecimal bytes in one of the following formats: ***nnnn.nnnn.nnnn*** or ***nn:nn:nn:nn:nn:nn***. For example, **0011.2233.4455** or **00:11:22:33:44:55**.

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

Related Documentation
- *Configuring the MAC Address on the Management Ethernet Interface*
- Configuring a Pseudowire Subscriber Logical Interface Device on page 325

mac-address (VLAN and Stacked VLAN Interfaces)

Syntax mac-address;

Hierarchy Level [edit interfaces *interface-name* auto-configure vlan-ranges authentication username-include],
[edit interfaces *interface-name* auto-configure stacked-vlan-ranges authentication username-include],

Release Information Statement introduced in Junos OS Release 10.0.

Description Specify that the client hardware address (chaddr) from the incoming DHCP discover packet be concatenated with the username during the subscriber authentication process.

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

Related Documentation
- Configuring VLAN Interface Username Information for AAA Authentication on page 22

mac-validate

Syntax mac-validate (loose | strict);

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

Release Information Statement introduced in Junos OS Release 9.3.

Description Enable IP and MAC address validation for static Ethernet and IP demux interfaces. Supported on MX Series routers only.

Options

loose—Forwards incoming packets when both the IP source address and the MAC source address match one of the trusted address tuples. Drops packets when the IP source address matches one of the trusted tuples, but the MAC address does not match the MAC address of the tuple. Continues to forward incoming packets when the source address of the incoming packet does not match any of the trusted IP addresses.

strict—Forwards incoming packets when both the IP source address and the MAC source address match one of the trusted address tuples. Drops packets when the MAC address does not match the tuple's MAC source address, or when IP source address of the incoming packet does not match any of the trusted IP addresses.

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

Related Documentation

- *MAC Address Validation on Static Ethernet Interfaces Overview*
- *Configuring MAC Address Validation on Static Demux Interfaces*

mac-validate (Dynamic IP Demux Interface)

Syntax mac-validate (loose | strict);

Hierarchy Level [edit dynamic-profiles *profile-name* interfaces demux0 unit *logical-unit-number* family inet]

Release Information Statement introduced in Junos OS Release 9.3.

Description Enable IP and MAC address validation for dynamic IP demux interfaces in a dynamic profile. Supported on MX Series routers only.

Options

loose—Forwards incoming packets when both the IP source address and the MAC source address match one of the trusted address tuples. Drops packets when the IP source address matches one of the trusted tuples, but the MAC address does not match the MAC address of the tuple. Continues to forward incoming packets when the source address of the incoming packet does not match any of the trusted IP addresses.

strict—Forwards incoming packets when both the IP source address and the MAC source address match one of the trusted address tuples. Drops packets when the MAC address does not match the tuple's MAC source address, or when IP source address of the incoming packet does not match any of the trusted IP addresses.

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

Related Documentation

- Configuring MAC Address Validation for Subscriber Interfaces on page 131

max-sessions (Dynamic PPPoE)

Syntax max-sessions *number*;

Hierarchy Level [edit dynamic-profiles *profile-name* interfaces demux0 unit *logical-unit-number* family pppoe],
[edit dynamic-profiles *profile-name* interfaces *interface-name* unit *logical-unit-number* family pppoe],
[edit dynamic-profiles *profile-name* interfaces interface-set *interface-set-name* pppoe-underlying-options]
[edit interfaces *interface-name* unit *logical-unit-number* family pppoe],
[edit interfaces *interface-name* unit *logical-unit-number* pppoe-underlying-options],
[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number* family pppoe],
[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number* pppoe-underlying-options]

Release Information Statement introduced in Junos OS Release 10.1.
Support for the **[edit ... family pppoe]** hierarchies introduced in Junos OS Release 11.2.
Support at the **[edit dynamic-profiles ... interfaces interface-set ... pppoe-underlying-options]** hierarchy level introduced in Junos OS Release 12.2.

Description Configure the maximum number of dynamic PPPoE logical interfaces that the router can activate on the underlying interface. The **max-sessions** value does not affect the maximum number of static PPPoE logical interfaces that can be configured on the underlying interface.

NOTE: The **[edit ... family pppoe]** hierarchies and the **[edit dynamic-profiles ... interfaces interface-set ... pppoe-underlying-options]** hierarchy level are supported only on MX Series routers with MPCs/MICs.

Options *number*—Maximum number of dynamic PPPoE logical interfaces (sessions) that the router can activate on the underlying interface. The default value is equal to the maximum number of PPPoE sessions supported on your routing platform. You can configure from 1 to the platform-specific default for your routing platform. Changing the **max-sessions** value has no effect on dynamic PPPoE logical interfaces that are already active.

For information about scaling values for PPPoE interfaces, access the *Subscriber Management Scaling Values (XLS)* spreadsheet from the Downloads box on the *Junos OS Subscriber Management* pathway page for the current release.

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

Related Documentation
- Limiting the Maximum Number of PPPoE Sessions on the Underlying Interface on page 189

- Defining Agent Circuit Identifier Interface Sets on page 31
- PPPoE Maximum Session Limit Overview on page 185
- Guidelines for Using PPPoE Maximum Session Limit from RADIUS on page 187
- *Juniper Networks VSAs Supported by the AAA Service Framework*
- *Configuring an Interface Set of Subscribers in a Dynamic Profile*
- Subscriber Interfaces and PPPoE Overview on page 141

max-sessions (PPPoE Service Name Tables)

Syntax max-sessions *number*;

Hierarchy Level [edit protocols pppoe service-name-tables *table-name* service *service-name*]

Release Information Statement introduced in Junos OS Release 10.2.

Description Configure the maximum number of active PPPoE sessions using either static or dynamic PPPoE interfaces that the router can establish with the specified named service, **empty** service, or **any** service entry in a PPPoE service name table. The router maintains a count of active PPPoE sessions for each service entry to determine when the maximum sessions limit has been reached.

The router uses the **max-sessions** value for a PPPoE service name table entry in conjunction with the **max-sessions** value configured for the PPPoE underlying interface, and with the maximum number of PPPoE sessions supported on your router. If your configuration exceeds any of these maximum session limits, the router is unable to establish the PPPoE session.

Options *number*—Maximum number of active PPPoE sessions that the router can establish with the specified PPPoE service name table entry, in the range 1 to the platform-specific maximum PPPoE sessions supported for your router. The default value is equal to the maximum number of PPPoE sessions supported on your routing platform.

For information about scaling values for PPPoE interfaces, access the *Subscriber Management Scaling Values (XLS)* spreadsheet from the Downloads box on the *Junos OS Subscriber Management* pathway page for the current release.

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

Related Documentation

- *Limiting the Number of Active PPPoE Sessions Established with a Specified Service Name*
- *Configuring PPPoE Service Name Tables*
- PPPoE Maximum Session Limit Overview on page 185
- *Configuring an Interface Set of Subscribers in a Dynamic Profile*
- Subscriber Interfaces and PPPoE Overview on page 141

max-sessions-vsa-ignore (Static and Dynamic Subscribers)

Syntax max-sessions-vsa-ignore;

Hierarchy Level [edit dynamic-profiles *profile-name* interfaces demux0 unit *logical-unit-number* family pppoe],
[edit dynamic-profiles *profile-name* interfaces *interface-name* unit *logical-unit-number* family pppoe],
[edit dynamic-profiles *profile-name* interfaces *interface-name* unit *logical-unit-number* pppoe-underlying-options],
[edit interfaces *interface-name* unit *logical-unit-number* family pppoe],
[edit interfaces *interface-name* unit *logical-unit-number* pppoe-underlying-options],
[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number* family pppoe],
[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number* pppoe-underlying-options]

Release Information Statement introduced in Junos OS Release 11.4.

Description Configure the router to ignore (clear) the value returned by RADIUS in the Max-Clients-Per-Interface Juniper Networks vendor-specific attribute (VSA) [26-143], and restore the PPPoE maximum session value on the underlying interface to the value configured in the CLI with the **max-sessions** statement. The PPPoE maximum session value specifies the maximum number of concurrent static or dynamic PPPoE logical interfaces (sessions) that the router can activate on the PPPoE underlying interface, or the maximum number of active static or dynamic PPPoE sessions that the router can establish with a particular service entry in a PPPoE service name table.

Default If you do not include the **max-sessions-vsa-ignore** statement, the maximum session value returned by RADIUS in the Max-Clients-Per-Interface VSA takes precedence over the PPPoE maximum session value configured with the **max-sessions** statement.

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

Related Documentation

- Limiting the Maximum Number of PPPoE Sessions on the Underlying Interface on page 189
- PPPoE Maximum Session Limit Overview on page 185
- Guidelines for Using PPPoE Maximum Session Limit from RADIUS on page 187
- *Juniper Networks VSAs Supported by the AAA Service Framework*
- *Configuring an Interface Set of Subscribers in a Dynamic Profile*
- Subscriber Interfaces and PPPoE Overview on page 141

mode (Dynamic Profiles)

Syntax mode loose;

Hierarchy Level [edit dynamic-profiles *profile-name* interfaces *interface-name* unit *logical-unit-number* family (inet) rpf-check]

Release Information Statement introduced in Junos OS Release 9.6.

Description Check whether the packet has a source address with a corresponding prefix in the routing table. If a corresponding prefix is not found, unicast reverse path forwarding (RPF) loose mode does not accept the packet. Unlike strict mode, loose mode does not check whether the interface expects to receive a packet with a specific source address prefix.

Default If you do not include this statement, unicast RPF is in strict mode.

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

Related Documentation

- *Configuring Unicast RPF*

mtu

Syntax mtu *bytes*;

Hierarchy Level [edit interfaces *interface-name*],
[edit interfaces *interface-name* unit *logical-unit-number* family *family*],
[edit interfaces interface-range *name*],
[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number* family *family*],
[edit logical-systems *logical-system-name* protocols l2circuit local-switching interface *interface-name* backup-neighbor *address*],
[edit logical-systems *logical-system-name* protocols l2circuit neighbor *address* interface *interface-name*],
[edit logical-systems *logical-system-name* protocols l2circuit neighbor *address* interface *interface-name* backup-neighbor *address*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols l2vpn interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols vpls],
[edit protocols l2circuit local-switching interface *interface-name* backup-neighbor *address*],
[edit protocols l2circuit neighbor *address* interface *interface-name*]
[edit protocols l2circuit neighbor *address* interface *interface-name* backup-neighbor *address*],
[edit routing-instances *routing-instance-name* protocols l2vpn interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols l2vpn site *site-name*],
[edit routing-instances *routing-instance-name* protocols vpls]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.
Support for Layer 2 VPNs and VPLS introduced in Junos OS Release 10.4.
Statement introduced in Junos OS Release 12.1X48 for PTX Series Packet Transport Routers.
Statement introduced in Junos OS Release 12.2 for ACX Series Universal Access Routers.
Support at the **[set interfaces interface-name unit logical-unit-number family ccc]** hierarchy level introduced in Junos OS Release 12.3R3 for MX Series routers.

Description Specify the maximum transmission unit (MTU) size for the media or protocol. The default MTU size depends on the device type. Changing the media MTU or protocol MTU causes an interface to be deleted and added again.

To route jumbo data packets on an integrated routing and bridging (IRB) interface or routed VLAN interface (RVI) on EX Series switches, you must configure the jumbo MTU size on the member physical interfaces of the VLAN that you have associated with the IRB interface or RVI, as well as on the IRB interface or RVI itself (the interface named irb or vlan, respectively).

CAUTION: For EX Series switches, setting or deleting the jumbo MTU size on an IRB interface or RVI while the switch is transmitting packets might cause packets to be dropped.

NOTE:
The MTU for an IRB interface is calculated by removing the Ethernet header overhead [6(DMAC)+6(SMAC)+2(EtherType)]. Because, the MTU is the lower value of the MTU configured on the IRB interface and the MTU configured on the IRB's associated bridge domain IFDs or IFLs, the IRB MTU is calculated as follows:

- In case of Layer 2 IFL configured with the **flexible-vlan-tagging** statement, the IRB MTU is calculated by including 8 bytes overhead (SVLAN+CVLAN).
- In case of Layer 2 IFL configured with the **vlan-tagging** statement, the IRB MTU is calculated by including a single VLAN 4 bytes overhead.

NOTE:

- If a packet whose size is larger than the configured MTU size is received on the receiving interface, the packet is eventually dropped. The value considered for MRU (maximum receive unit) size is also the same as the MTU size configured on that interface.
- Not all devices allow you to set an MTU value, and some devices have restrictions on the range of allowable MTU values. You cannot configure an MTU for management Ethernet interfaces (fxp0, em0, or me0) or for loopback, multilink, and multicast tunnel devices.
- On ACX Series routers, you can configure the protocol MTU by including the **mtu** statement at the **[edit interfaces *interface-name* unit *logical-unit-number* family inet]** or **[edit interfaces *interface-name* unit *logical-unit-number* family inet6]** hierarchy level.
    - If you configure the protocol MTU at any of these hierarchy levels, the configured value is applied to all families that are configured on the logical interface.
    - If you are configuring the protocol MTU for both **inet** and **inet6** families on the same logical interface, you must configure the same value for both the families. It is not recommended to configure different MTU size values for **inet** and **inet6** families that are configured on the same logical interface.

For more information about configuring MTU for specific interfaces and router or switch combinations, see *Configuring the Media MTU*.

Options ***bytes***—MTU size.

Range: 256 through 9192 bytes, 256 through 9216 (EX Series switch interfaces), 256 through 9500 bytes (Junos OS 12.1X48R2 for PTX Series routers)

Default: 1500 bytes (INET, INET6, and ISO families), 1448 bytes (MPLS), 1514 bytes (EX Series switch interfaces)

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

Related Documentation

- *Configuring Gigabit Ethernet Interfaces (CLI Procedure)*
- *Configuring the Media MTU*
- *Configuring the MTU for Layer 2 Interfaces*
- *Setting the Protocol MTU*

nas-port-extended-format (Access Profile)

Syntax

```
nas-port-extended-format {
  adapter-width width;
  ae-width width;
  port-width width;
  slot-width width;
  stacked-vlan-width width;
  vlan-width width;
  atm {
    adapter-width width;
    port-width width:
    slot-width width;
    vci-width width:
    vpi-width width;
  }
}
```

Hierarchy Level

[edit access profile *profile-name* radius options]

Release Information

Statement introduced in Junos OS Release 9.1.
Statement introduced in Junos OS Release 9.1 for EX Series switches.
Option **ae-width** introduced in Junos OS Release 12.1.
Option **stacked** introduced in Junos OS Release 12.3.
Option **atm** introduced in Junos OS Release 12.3R3 and supported in later 12.3R*x* releases.
Option **atm** supported in Junos OS Release 13.2 and later releases. (Not supported in Junos OS Release 13.1.)

Description

In an access profile, configure the RADIUS client to use the extended format for RADIUS attribute 5 (NAS-Port) and specify the width of the fields in the NAS-Port attribute. You can use the same access profile to configure the NAS-Port extended format for Ethernet subscribers and ATM subscribers.

Options

adapter-width ***width***—Number of bits in the adapter field.

ae-width ***width***—Number of bits in the aggregated Ethernet identifier field.

port-width ***width***—Number of bits in the port field.

slot-width ***width***—Number of bits in the slot field.

stacked—Include stacked VLAN IDs, in addition to VLAN IDs, in the NAS-Port extended format.

stacked-vlan-width ***width***—Number of bits in the SVLAN ID field.

vlan-width ***width***—Number of bits in the VLAN ID field.

atm—Configure the NAS-Port extended format for ATM subscribers; options include:

- **adapter-width** ***width***—Number of bits in the adapter field.
- **port-width** ***width***—Number of bits in the port field.

- **slot-width** ***width***—Number of bits in the slot field.
- **vci-width** ***width***—Number of bits in the ATM virtual circuit identifier (VCI) field.
- **vpi-width** ***width***—Number of bits in the ATM virtual path identifier (VPI) field.

NOTE: Each field can be 0 through 32 bits wide; however, the total of the widths of all fields must not exceed 32 bits, or the configuration fails.

The router may truncate the values of individual fields depending on the bit width you specify.

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

Related Documentation

- Configuring RADIUS Server Options for Subscriber Access on page 364
- *Configuring RADIUS Server Parameters for Subscriber Access*

nas-port-extended-format (Interfaces)

Syntax

```
nas-port-extended-format {
  adapter-width width;
  ae-width width;
  port-width width;
  slot-width width;
  stacked;
  stacked-vlan-width width;
  vci-width width:
  vlan-width width;
  vpi-width width;
}
```

Hierarchy Level [edit interfaces *interface-name* radius-options nas-port-options *nas-port-options-name*]

Release Information Statement introduced in Junos OS Release 12.3.
Options **vci-width** and **vpi-width** introduced in Junos OS Release 12.3R3 and supported in later 12.3R*x* releases.
Options **vci-width** and **vpi-width** supported in Junos OS Release 13.2 and later releases. (Not supported in Junos OS Release 13.1.)

Description Configure the RADIUS client to use the extended format for RADIUS attribute 5 (NAS-Port) and specify the width of the fields in the NAS-Port attribute.

Options **adapter-width *width***—Number of bits in the adapter field.

ae-width *width*—Number of bits in the aggregated Ethernet identifier field.

port-width *width*—Number of bits in the port field.

slot-width *width*—Number of bits in the slot field.

stacked—Include stacked VLAN IDs, in addition to VLAN IDs, in the NAS-Port extended format.

stacked-vlan-width *width*—Number of bits in the SVLAN ID field.

vci-width *width*—Number of bits in the ATM virtual circuit identifier (VCI) field.

vlan-width *width*—Number of bits in the VLAN ID field.

vpi-width *width*—Number of bits in the ATM virtual path identifier (VPI) field.

NOTE: Each field can be 0 through 32 bits wide; however, the total of the widths of all fields must not exceed 32 bits, or the configuration fails.

The router may truncate the values of individual fields depending on the bit width you specify.

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

Related Documentation

- *Configuring RADIUS NAS-Port Options for Subscriber Access per Physical Interface, VLAN, or Stacked VLAN*
- *Guidelines for Configuring RADIUS NAS-Port Options for Subscriber Access per Physical Interface, VLAN, or Stacked VLAN*

nd-override-preferred-src

Syntax nd-override-preferred-src;

Hierarchy Level [edit system]

Release Information Statement introduced in Junos OS Release 13.3

Description Configure the router to override the default configuration and use the appropriate address based on destination address scope for the source address for Neighbor Solicitation/Neighbor Advertisement (NS/NA) for unnumbered interfaces.

Default The router uses the preferred source address, if configured, as source for NS/NA for unnumbered interfaces. If no preferred source address is configured, the router uses the appropriate address based on destination address scope.

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

Related Documentation

- unnumbered-address on page 586

no-gratuitous-arp-request

Syntax no-gratuitous-arp-request;

Hierarchy Level [edit interfaces *interface-name*]

Release Information Statement introduced in Junos OS Release 9.6 for EX Series switches.
Statement introduced in Junos OS Release 12.2 for ACX Series Universal Access Routers.

Description For Ethernet interfaces and pseudowire logical interfaces, do not respond to gratuitous ARP requests.

Default Gratuitous ARP responses are enabled on all Ethernet interfaces.

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

Related Documentation

- *Configuring Gratuitous ARP*
- *gratuitous-arp-reply*

no-keepalives (Dynamic Profiles)

Syntax no-keepalives;

Hierarchy Level [edit dynamic-profiles *profile-name* interfaces *interface-name* unit *logical-unit-number*],
[edit dynamic-profiles *profile-name* interfaces pp0 unit "$junos-interface-unit"]

Release Information Statement introduced before Junos OS Release 7.4.
Support of the **[edit dynamic-profiles *profile-name*]** hierarchy level introduced in Junos OS Release 9.5.
Support of the **[edit dynamic-profiles *profile-name* interfaces pp0 unit "$junos-interface-unit"]** hierarchy level introduced in Junos OS Release 10.1.

Description Disable the sending of keepalives.

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

Related Documentation

- *Dynamic Profiles Overview*
- *Configuring Dynamic Authentication for PPP Subscribers*

no-vlan-id-validate

Syntax no-vlan-id-validate;

Hierarchy Level [edit logical-systems *logical-system-name* protocols l2circuit neighbor *address* interface *interface-name*],
[edit protocols l2circuit neighbor *address* interface *interface-name*]

Release Information Statement introduced in Junos OS Release 13.1.

Description Uniquely identify a Layer 2 circuit for either a standard pseudowire or a redundant pseudowire.

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

Related Documentation

- *Configuring Interfaces for Layer 2 Circuits*
- Pseudowire Subscriber Logical Interfaces Overview on page 321
- Configuring a Pseudowire Subscriber Logical Interface on page 323
- Configuring Layer 2 Circuit Signaling for Pseudowire Subscriber Logical Interfaces on page 328

oam-on-svlan (Ethernet Interfaces)

Syntax oam-on-svlan;

Hierarchy Level [edit interfaces *interface-name*]

Release Information Statement introduced in Junos OS Release 13.1.

Description On MX Series routers with MPC/MIC interfaces, enable propagation of the Ethernet IEEE 802.1ag Operation, Administration, and Maintenance (OAM) state of a static single-tagged service VLAN (S-VLAN) logical interface to the dynamic or static double-tagged customer VLAN (C-VLAN) logical interface and associated subscriber interfaces configured on the S-VLAN. The static S-VLAN logical interface must be configured with Ethernet OAM connectivity fault management (CFM) on a Gigabit Ethernet, 10-Gigabit Ethernet, or aggregated Ethernet physical interface. The C-VLAN logical interface must have the same S-VLAN (outer) tag as the S-VLAN logical interface.

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

Related Documentation

- Configuring Ethernet OAM Support for Service VLANs with Double-Tagged Customer VLANs on page 62
- Ethernet OAM Support for Service VLANs Overview on page 59

option-18 (Interface-ID for DHCPv6 Autosense VLANs)

Syntax option-18;

Hierarchy Level [edit interfaces *interface-name* auto-configure vlan-ranges authentication username-include],
[edit interfaces *interface-name* auto-configure stacked-vlan-ranges authentication username-include]

Release Information Statement introduced in Junos OS Release 13.2.

Description Specify that Option 18 (Interface-ID) information received in the innermost DHCPv6 Relay-Forward message header is concatenated with the username during the subscriber authentication process.

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

Related Documentation

- Configuring VLAN Interface Username Information for AAA Authentication on page 22
- *Inserting DHCPv6 Interface-ID Option (Option 18) In DHCPv6 Packets*
- *Creating Unique Usernames for DHCP Clients*
- Using DHCP Option 18 and Option 37 in Authentication Usernames for DHCPv6 Autosense VLANs on page 24
- **option-37 (Relay Agent Remote-ID for DHCPv6 Autosense VLANs) on page 525**

option-37 (Relay Agent Remote-ID for DHCPv6 Autosense VLANs)

Syntax option-37;

Hierarchy Level [edit interfaces *interface-name* auto-configure vlan-ranges authentication username-include],
[edit interfaces *interface-name* auto-configure stacked-vlan-ranges authentication username-include]

Release Information Statement introduced in Junos OS Release 13.2.

Description Specify that Option 37 (DHCPv6 Relay Agent Remote-ID) information, received in the innermost DHCPv6 Relay-Forward message header, is concatenated with the username during the subscriber authentication process.

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

Related Documentation

- Configuring VLAN Interface Username Information for AAA Authentication on page 22
- *relay-agent-remote-id*
- *Creating Unique Usernames for DHCP Clients*
- *Inserting DHCPv6 Interface-ID Option (Option 18) In DHCPv6 Packets*
- Using DHCP Option 18 and Option 37 in Authentication Usernames for DHCPv6 Autosense VLANs on page 24
- **option-18 (Interface-ID for DHCPv6 Autosense VLANs) on page 524**

option-82

Syntax option-82 <circuit-id> <remote-id>;

Hierarchy Level [edit interfaces *interface-name* auto-configure vlan-ranges authentication username-include], [edit interfaces *interface-name* auto-configure stacked-vlan-ranges authentication username-include]

Release Information Statement introduced in Junos OS Release 10.0.
Options **circuit-id** and **remote-id** introduced in Junos OS Release 11.4.

Description Specify that the option 82 information from the client PDU is concatenated with the username during the subscriber authentication process.

For autosense VLANs, you can additionally specify Option 82 suboption information that is concatenated with the username. You can specify either both or neither of the Agent Circuit ID (suboption 1) and Agent Remote ID (suboption 1). If you specify both, the Agent Circuit ID is supplied first, followed by a delimiter, and then the Agent Remote ID. If you specify that neither suboption is supplied, the raw payload of Option 82 from the PDU is concatenated to the username.

NOTE: **The option 82 value used in creating the username is based on the option 82 value that is encoded in the incoming DHCP discover packet. The use of suboptions is supported for DHCPv4 only.**

Options **none**—Use the raw payload of Option 82 from the PDU.

circuit-id—(Optional) Use the Agent Circuit ID suboption (suboption 1).

remote-id—(Optional) Use the Agent Remote ID suboption (suboption 2).

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

Related Documentation

- Configuring VLAN Interface Username Information for AAA Authentication on page 22
- Using DHCP Option 82 Suboptions in Authentication Usernames for Autosense VLANs on page 23

output (Dynamic Service Sets)

Syntax

```
output {
  service-set service-set-name {
  service-filter filter-name;
  }
}
```

Hierarchy Level [edit dynamic-profiles *profile-name* interfaces *interface-name* unit *logical-unit-number* family *family* service],
[edit dynamic-profiles *profile-name* interfaces pp0 unit "$junos–interface–unit" family *family* service]

Release Information Statement introduced in Junos OS Release 9.5.
Support of the **[edit dynamic-profiles *profile-name* interfaces pp0 unit "$junos-interface-unit" family *family* service]** hierarchy level introduced in Junos OS Release 10.1.

Description Define the output service sets and filters to be applied to traffic by a dynamic profile. Only the Internet Protocol version 4 (IPv4) protocol family is currently supported for dynamic PPPoE logical interfaces.

The remaining statement is explained separately.

Options ***service-set-name***—Name of the service set.

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

Related Documentation

- *Dynamic Service Sets Overview*
- *Associating Service Sets with Interfaces in a Dynamic Profile*

output-traffic-control-profile (Dynamic CoS Definition)

Syntax output-traffic-control-profile (*profile-name* | $junos-cos-traffic-control-profile);

Hierarchy Level [edit dynamic-profiles *profile-name* class-of-service interfaces *interface-name* unit *logical-unit-number*]

Release Information Statement introduced in Junos OS Release 9.2.
Variable **$junos-cos-traffic-control-profile** introduced in Junos OS Release 11.2.

Description Apply an output traffic scheduling and shaping profile to the logical interface.

Options ***profile-name***—Name of the traffic-control profile to be applied to this interface

$junos-cos-traffic-control-profile—Variable for the traffic-control profile that is specified for the logical interface. The variable is replaced with the traffic-control profile when the subscriber is authenticated at login.

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

Related Documentation

- *Guidelines for Configuring Dynamic CoS for Subscriber Access*
- *Applying Traffic Shaping and Scheduling to a Subscriber Interface in a Dynamic Profile*
- *Using the CLI to Modify Traffic-Control Profiles That Are Currently Applied to Subscribers*
- traffic-control-profiles on page 568

output-vlan-map (Dynamic Interfaces)

Syntax

```
output-vlan-map {
  inner-tag-protocol-id tpid;
  inner-vlan-id number;
  (pop | swap);
  tag-protocol-id tpid;
  vlan-id number;
}
```

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

Release Information Statement introduced in Junos OS Release 10.4.

Description For dynamic interfaces, define the rewrite profile to be applied to outgoing frames on this logical interface.

The statements are explained separately.

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

Related Documentation

- *Stacking and Rewriting VLAN Tags for the Layer 2 Wholesale Solution*

override

Syntax override tag *vlan-tag* dynamic-profile *profile name*;

Hierarchy Level [edit interfaces *interface-name* auto-configure vlan-ranges],
[edit interfaces *interface-name* auto-configure stacked-vlan-ranges]

Release Information Statement introduced in Junos OS Release 11.2.

Description Override dynamic profile assignment to individual VLANs that are already part of a previously defined VLAN range and dynamic profile.

Options ***vlan-tag***—VLAN tag that you want to override.

profile-name—Name of the dynamic profile that you want to use when overriding the specified VLAN tag.

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

Related Documentation

- Overriding the Dynamic Profile Used for an Individual VLAN on page 14
- *Configuring VLAN Ranges for Use with Dynamic Profiles*

packet-types (Dynamic VLAN Authentication)

Syntax packet-types [*packet-types*]

Hierarchy Level [edit interfaces *interface-name* auto-configure vlan-ranges authentication],
[edit interfaces *interface-name* auto-configure stacked-vlan-ranges authentication]

Release Information Statement introduced in Junos OS Release 14.1.

Description Specify one or more packet types to trigger authentication of an auto-configured dynamic VLAN. The packet types must be a subset of the packet types configured in the VLAN dynamic profile to trigger creation of the dynamic VLAN.

Options ***packet-type***—One or more of the following packet types that triggers VLAN authentication:

- **any**—Any packet type.
- **dhcp-v4**—IPv4 DHCP packet type.
- **dhcp-v6**—IPv6 DHCP packet type.
- **inet**—IPv4 Ethernet and ARP packet type.
- **inet6**—IPv6 Ethernet packet type.
- **pppoe**—Point-to-Point Protocol over Ethernet packet type.

NOTE: The **pppoe** VLAN Ethernet packet type option is supported only for MIC and MPC interfaces on MX Series routers.

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

Related Documentation

- Configuring Subscriber Packet Types to Trigger VLAN Authentication on page 22
- *Subscriber Packet Type Authentication Triggers for Dynamic VLANs*

pap (Dynamic PPP)

Syntax pap;

Hierarchy Level [edit dynamic-profiles *profile-name* interfaces pp0 unit "$junos-interface-unit" ppp-options], [edit dynamic-profiles *profile-name* interfaces "$junos-interface-ifd-name" unit "$junos-interface-unit" ppp-options]

Release Information Statement introduced in Junos OS Release 9.5.
Support at the **[edit dynamic-profiles *profile-name* interfaces "$junos-interface-ifd-name" unit "$junos-interface-unit" ppp-options]** hierarchy level introduced in Junos OS Release 12.2.

Description Specify PAP authentication in a PPP dynamic profile.

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

Related Documentation

- *Dynamic Profiles Overview*
- *Configuring Dynamic Authentication for PPP Subscribers*
- *Attaching Dynamic Profiles to Static PPP Subscriber Interfaces*
- *Applying PPP Attributes to L2TP LNS Subscribers Per Inline Service Interface*

passive (CHAP)

Syntax

passive;

Hierarchy Level

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

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Do not challenge the peer, but respond if challenged. If you omit this statement from the configuration, the interface always challenges its peer.

For ATM2 IQ interfaces only, you can configure CHAP on the logical interface unit if the logical interface is configured with one of the following PPP over ATM encapsulation types:

- **atm-ppp-llc**—PPP over AAL5 LLC encapsulation.
- **atm-ppp-vc-mux**—PPP over AAL5 multiplex encapsulation.

Required Privilege Level

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

Related Documentation

- *Configuring Passive Mode*
- *Junos OS Administration Library for Routing Devices*

pop (Dynamic VLANs)

Syntax

pop;

Hierarchy Level

[edit dynamic-profiles *profile-name* interfaces *interface-name* unit *logical-unit-number* output-vlan-map]

Release Information

Statement introduced in Junos OS Release 10.4.

Description

For dynamic VLAN interfaces, specify the VLAN rewrite operation to remove a VLAN tag from the top of the VLAN tag stack. The outer VLAN tag of the frame is removed.

Required Privilege Level

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

Related Documentation

- *Removing a VLAN Tag*
- *Stacking and Rewriting VLAN Tags for the Layer 2 Wholesale Solution*

post-service-filter (Dynamic Service Sets)

Syntax post-service-filter *filter-name*;

Hierarchy Level [edit dynamic-profiles *profile-name* interfaces *interface-name* unit *logical-unit-number* family *family* service input],
[edit dynamic-profiles *profile-name* interfaces pp0 unit "$junos–interface–unit" family *family* service input]

Release Information Statement introduced in Junos OS Release 9.5.
Support at the **[edit dynamic-profiles *profile-name* interfaces pp0 unit "$junos-interface-unit" family *family* service input]** hierarchy level introduced in Junos OS Release 10.1.

Description Define the filter to be applied to traffic after service processing. The filter is applied only if a service set is configured and selected. You can configure a postservice filter on the input side of the interface only. Only the Internet Protocol version 4 (IPv4) protocol family is currently supported for dynamic PPPoE logical interfaces.

Options ***filter-name***—Identifier for the post-service filter.

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

Related Documentation

- *Dynamic Service Sets Overview*
- *Associating Service Sets with Interfaces in a Dynamic Profile*

pp0 (Dynamic PPPoE)

Syntax

```
pp0 {
  unit logical-unit-number {
    keepalives interval seconds;
    no-keepalives;
    pppoe-options {
      underlying-interface interface-name;
      server;
    }
    ppp-options {
      authentication [ authentication-protocols ];
      chap {
        challenge-length minimum minimum-length maximum maximum-length;
      }
      pap;
    }
    family inet {
      unnumbered-address interface-name;
      address address;
      service {
        input {
          service-set service-set-name {
            service-filter filter-name;
          }
          post-service-filter filter-name;
        }
        output {
          service-set service-set-name {
            service-filter filter-name;
          }
        }
      }
      filter {
        input filter-name {
          precedence precedence;
        }
        output filter-name {
          precedence precedence;
        }
      }
    }
  }
}
```

Hierarchy Level [edit dynamic-profiles *profile-name* interfaces]

Release Information Statement introduced in Junos OS Release 10.1.

Description Configure the dynamic PPPoE logical interface in a dynamic profile. When the router creates a dynamic PPPoE logical interface on an underlying Ethernet interface configured with PPPoE (**ppp-over-ether**) encapsulation, it uses the information in the dynamic profile to determine the properties of the dynamic PPPoE logical interface.

The remaining statements are explained separately.

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

Related Documentation

- Configuring a Basic PPPoE Dynamic Profile on page 146
- Configuring a PPPoE Dynamic Profile with Additional Options on page 148
- *Configuring Dynamic Authentication for PPP Subscribers*
- For information about creating static PPPoE interfaces, see *Configuring PPPoE*

ppp-options

Syntax

```
ppp-options {
  authentication [ authentication-protocols ];
  chap {
    access-profile name;
    challenge-length minimum minimum-length maximum maximum-length;
    default-chap-secret name;
    local-name name;
    passive;
  }
  compression {
    acfc;
    pfc;
  }
  dynamic-profile profile-name;
  initiate-ncp (ip | ipv6 | dual-stack-passive)
  lcp-max-conf-req number
  lcp-restart-timer milliseconds;
  loopback-clear-timer seconds;
  ncp-max-conf-req number
  ncp-restart-timer milliseconds;
  on-demand-ip-address
  pap {
    access-profile name;
    default-pap-password password;
    local-name name;
    local-password password;
    passive;
  }
}
```

Hierarchy Level

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

Release Information

Statement introduced before Junos OS Release 7.4.

Description

On interfaces with PPP encapsulation, configure PPP-specific interface properties.

For ATM2 IQ interfaces only, you can configure CHAP on the logical interface unit if the logical interface is configured with one of the following PPP over ATM encapsulation types:

- **atm-ppp-llc**—PPP over AAL5 LLC encapsulation.
- **atm-ppp-vc-mux**—PPP over AAL5 multiplex encapsulation.

BEST PRACTICE: On inline service (si) interfaces for L2TP, only the **chap** and **pap** statements are typically used for subscriber management. We recommend that you leave the other statements subordinate to

ppp-options—including those subordinate to **chap** and **pap**—at their default values.

The remaining statements are explained separately.

Required Privilege Level

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

Related Documentation

- *Configuring the PPP Challenge Handshake Authentication Protocol*
- *Applying PPP Attributes to L2TP LNS Subscribers Per Inline Service Interface*

ppp-options (Dynamic PPP)

Syntax

```
ppp-options {
  authentication [ authentication-protocols ];
  chap {
    challenge-length minimum minimum-length maximum maximum-length;
  }
  initiate-ncp (ip | ipv6 | dual-stack-passive)
  on-demand-ip-address;
  pap;
}
```

Hierarchy Level

[edit dynamic-profiles *profile-name* interfaces pp0 unit “$junos-interface-unit”],
[edit dynamic-profiles *profile-name* interfaces "$junos-interface-ifd-name" unit
“$junos-interface-unit”]

Release Information

Statement introduced in Junos OS Release 9.5.
Support at the **[edit dynamic-profiles *profile-name* interfaces "$junos-interface-ifd-name" unit “$junos-interface-unit”]** hierarchy level introduced in Junos OS Release 12.2.

Description

Configure PPP-specific interface properties in a dynamic profile.

The remaining statements are explained separately.

Required Privilege Level

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

Related Documentation

- *Dynamic Profiles Overview*
- *Configuring Dynamic Authentication for PPP Subscribers*
- *Attaching Dynamic Profiles to Static PPP Subscriber Interfaces*
- *Applying PPP Attributes to L2TP LNS Subscribers Per Inline Service Interface*

ppp-subscriber-services

Syntax ppp-subscriber-services (disable | enable);

Hierarchy Level [edit chassis]

Release Information Statement introduced in Junos OS Release 10.2.

Description Enable dynamic PPP subscriber services on non-PPPoE interfaces on certain PICs.

NOTE: When you include this statement, the relevant PICs restart. This action disrupts subscribers already logged in through those PICs. You can confirm completion of the restart by issuing the **show chassis pic fpc-slot** ***slot-number*** **pic-slot** ***slot-number*** command.

Options **disable**—Disable subscriber services.

enable—Enable subscriber services.

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

Related Documentation

- *show chassis pic*
- Attaching Dynamic Profiles to MLPPP Bundles on page 307
- For hardware requirements, see Hardware Requirements for PPP Subscriber Services on Non-Ethernet Interfaces on page 306

pppoe-options

Syntax

pppoe-options {
  access-concentrator *name*;
  auto-reconnect *seconds*;
  (client | server);
  service-name *name*;
  underlying-interface *interface-name*;
}

Hierarchy Level

[edit interfaces pp0 unit *logical-unit-number*],
[edit logical-systems *logical-system-name* interfaces pp0 unit *logical-unit-number*]

Release Information

Statement introduced before Junos OS Release 7.4.
client Statement introduced in Junos OS Release 8.5.
server Statement introduced in Junos OS Release 8.5.

Description

For J Series Services Routers, M120 Multiservice Edge Routers, M320 Multiservice Edge Service Routers, and MX Series Universal Edge Routers with PPP over Ethernet interfaces, configure PPP over Ethernet-specific interface properties.

The remaining statements are explained separately.

Required Privilege Level

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

Related Documentation

- *Configuring a PPPoE Interface*

pppoe-options (Dynamic PPPoE)

Syntax

pppoe-options {
  underlying-interface *interface-name*;
  server;
}

Hierarchy Level

[edit dynamic-profiles *profile-name* interfaces pp0 unit “$junos-interface-unit”]

Release Information

Statement introduced in Junos OS Release 10.1.

Description

Configure the underlying interface and PPPoE server mode for a dynamic PPPoE logical interface in a dynamic profile.

The remaining statements are explained separately.

Required Privilege Level

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

Related Documentation

- Configuring a Basic PPPoE Dynamic Profile on page 146
- Configuring Dynamic PPPoE Subscriber Interfaces Using Dynamic Profiles on page 145

pppoe-underlying-options (Static and Dynamic Subscribers)

Syntax

```
pppoe-underlying-options {
  access-concentrator name;
  dynamic-profile profile-name;
  direct-connect
  duplicate-protection;
  max-sessions number;
  max-sessions-vsa-ignore;
  service-name-table table-name;
  short-cycle-protection <lockout-time-min minimum-seconds> <lockout-time-max
    maximum-seconds> <filter [aci]>;
}
```

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

Release Information Statement introduced in Junos OS Release 10.0.

Description Configure PPPoE-specific interface properties for the underlying interface on which the router creates a static or dynamic PPPoE logical interface. The underlying interface must be configured with PPPoE (**ppp-over-ether**) encapsulation.

The remaining statements are explained separately.

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

Related Documentation

- *Configuring PPPoE* (for static interfaces)
- Configuring an Underlying Interface for Dynamic PPPoE Subscriber Interfaces on page 156
- *Assigning a Service Name Table to a PPPoE Underlying Interface*

pppoe-underlying-options (Dynamic VLAN Interface Sets)

Syntax

```
pppoe-underlying-options {
  max-sessions number;
}
```

Hierarchy Level [edit dynamic-profiles *profile-name* interfaces interface-set "$junos-interface-set-name"]

Release Information Statement introduced in Junos OS Release 12.2.

Description Configure PPPoE-specific interface properties in the dynamic profile that defines the agent circuit identifier (ACI) interface set. An ACI interface set is a logical collection of subscriber interfaces that originate at the same household or on the same access-loop port. Configuring PPPoE-specific interface properties for an ACI interface set enables you to apply these attributes to all subscribers on a per-household basis.

The remaining statement is explained separately.

NOTE: **When you configure PPPoE-specific interface properties for an ACI interface set, only the max-sessions statement is currently supported.**

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

Related Documentation

- Configuring Dynamic VLANs Based on Agent Circuit Identifier Information on page 29
- Agent Circuit Identifier-Based Dynamic VLANs Components Overview on page 27

precedence

Syntax

precedence *precedence*;

Hierarchy Level

[edit dynamic-profiles *profile-name* interfaces *interface-name* unit *logical-unit-number* family *family* filter input *filter-name*],
[edit dynamic-profiles *profile-name* interfaces *interface-name* unit *logical-unit-number* family *family* filter output *filter-name*],
[edit dynamic-profiles *profile-name* interfaces demux0 unit *logical-unit-number* family *family* filter input *filter-name*],
[edit dynamic-profiles *profile-name* interfaces demux0 unit *logical-unit-number* family *family* filter output *filter-name*],
[edit dynamic-profiles *profile-name* interfaces pp0 unit "$junos–interface–unit" family *family* filter input *filter-name*],
[edit dynamic-profiles *profile-name* interfaces pp0 unit "$junos–interface–unit" family *family* filter output *filter-name*]

Release Information

Statement introduced in Junos OS Release 9.3.

The **[edit dynamic-profiles *profile-name* interfaces pp0 unit "$junos-interface-unit" family inet filter input *filter-name*]** hierarchy level and **[edit dynamic-profiles *profile-name* interfaces pp0 unit "$junos-interface-unit" family inet filter output *filter-name*]** hierarchy level introduced in Junos OS Release 10.1.

Description

Apply a precedence to a dynamic filter. Only the Internet Protocol version 4 (IPv4) protocol family is currently supported for dynamic PPPoE logical interfaces.

Options

precedence—Precedence value for the filter. The lower the precedence value, the higher the precedence.

Range: 0 through 250

Default: 0

Required Privilege Level

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

Related Documentation

- *Firewall Filters Overview*
- *Understanding Dynamic Firewall Filters*
- *Classic Filters Overview*
- *Fast Update Filters Overview*
- *Basic Classic Filter Syntax*
- *Basic Fast Update Filter Syntax*

profile (Access)

Syntax

```
profile profile-name {
  accounting {
    address-change-immediate-update
    accounting-stop-on-access-deny;
    accounting-stop-on-failure;
    ancp-speed-change-immediate-update;
    coa-immediate-update;
    coa-no-override service-class-attribute;
    duplication;
    duplication-vrf {
      access-profile-name profile-name;
      vrf-name vrf-name;
    }
    immediate-update;
    order [ accounting-method ];
    send-acct-status-on-config-change;
    statistics (time | volume-time);
    update-interval minutes;
    wait-for-acct-on-ack;
  }
  authentication-order [ authentication-methods ];
  client client-name {
    chap-secret chap-secret;
    group-profile profile-name;
    ike {
      allowed-proxy-pair {
        remote remote-proxy-address local local-proxy-address;
      }
      pre-shared-key (ascii-text character-string | hexadecimal hexadecimal-digits);
      ike-policy policy-name;
      interface-id string-value;
    }
    l2tp {
      aaa-access-profile profile-name;
      interface-id interface-id;
      lcp-renegotiation;
      local-chap;
      maximum-sessions-per-tunnel number;
      multilink {
        drop-timeout milliseconds;
        fragment-threshold bytes;
      }
      ppp-authentication (chap | pap);
      ppp-profile profile-name;
      shared-secret shared-secret;
    }
    pap-password pap-password;
    ppp {
      cell-overhead;
      encapsulation-overhead bytes;
      framed-ip-address ip-address;
      framed-pool framed-pool;
```

```
      idle-timeout seconds;
      interface-id interface-id;
      keepalive seconds;
      primary-dns primary-dns;
      primary-wins primary-wins;
      secondary-dns secondary-dns;
      secondary-wins secondary-wins;
    }
    user-group-profile profile-name;
  }
  domain-name-server;
  domain-name-server-inet;
  domain-name-server-inet6;
  preauthentication-order preauthentication-method;
  provisioning-order (gx-plus | jsrc);
  radius {
    accounting-server [ ip-address ];
    attributes {
      exclude {
        ...
      }
      ignore {
        framed-ip-netmask;
        input-filter;
        logical-system:routing-instance;
        output-filter;
      }
    }
    authentication-server [ ip-address ];
    options {
      accounting-session-id-format (decimal | description);
      calling-station-id-delimiter delimiter-character;
      calling-station-id-format {
        agent-circuit-id;
        agent-remote-id;
        interface-description;
        nas-identifier;
      }
      client-accounting-algorithm (direct | round-robin);
      client-authentication-algorithm (direct | round-robin);
      coa-dynamic-variable-validation;
      ethernet-port-type-virtual;
      interface-description-format {
        exclude-adapter;
        exclude-sub-interface;
      }
      juniper-dsl-attributes;
      nas-identifier identifier-value;
      nas-port-extended-format {
        adapter-width width;
        ae-width width;
        port-width width;
        slot-width width;
        stacked-vlan-width width;
        vlan-width width;
        atm {
```



```
                adapter-width width;
                port-width width:
                slot-width width;
                vci-width width:
                vpi-width width;
            }
        }
        nas-port-id-delimiter delimiter-character;
        nas-port-id-format {
            agent-circuit-id;
            agent-remote-id;
            interface-description;
            nas-identifier;
        }
        nas-port-type {
            ethernet {
                port-type;
            }
        }
        revert-interval interval;
        vlan-nas-port-stacked-format;
    }
    preauthentication-server ip-address;
  }
  radius-server server-address {
    accounting-port port-number;
    accounting-retry number;
    accounting-timeout seconds;
    port port-number;
    retry attempts;
    routing-instance routing-instance-name;
    secret password;
    max-outstanding-requests value;
    source-address source-address;
    timeout seconds;
  }
  service {
    accounting-order (activation-protocol | radius);
  }
  session-options {
    client-idle-timeout minutes;
    client-session-timeoutminutes;
  }
}
```

Hierarchy Level [edit access]

Release Information Statement introduced before Junos OS Release 7.4.

Description Configure PPP CHAP, or a profile and its subscriber access, L2TP, or PPP properties.

Options *profile-name*—Name of the profile.

For CHAP, the name serves as the mapping between peer identifiers and CHAP secret keys. This entity is queried for the secret key whenever a CHAP challenge or response is received.

The remaining statements are explained separately.

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

Related Documentation

- *Configuring the PPP Authentication Protocol*
- *Configuring Access Profiles for L2TP or PPP Parameters*
- *Configuring L2TP Properties for a Client-Specific Profile*
- *Configuring an L2TP LNS with Inline Service Interfaces*
- *Configuring PPP Properties for a Client-Specific Profile*
- *Configuring Service Accounting with JSRC*
- *AAA Service Framework Overview*
- *show network-access aaa statistics*
- *clear network-access aaa statistics*

proxy-arp

Syntax proxy-arp;

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

Release Information Statement introduced in Junos OS Release 9.5.

Description For Ethernet interfaces only, configure the router to respond to any ARP request, as long as the router has an active route to the target address of the ARP request.

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

Related Documentation

- *Configuring Restricted and Unrestricted Proxy ARP*
- *Configuring Gratuitous ARP*

ps0 (Pseudowire Subscriber Interfaces)

Syntax

```
ps0 {
  anchor-point lt-device;
  mtu bytes;
  mac mac-address;
  no-gratuitous-arp-request;
  (flexible-vlan-tagging | stacked-vlan-tagging | untagged | vlan-tagging);
}
```

Hierarchy Level [edit logical-systems transport-ls interfaces]

Release Information Statement introduced in Junos OS Release 13.1.

Description Configure the pseudowire logical device.

The remaining statements are explained separately.

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

Related Documentation

- Pseudowire Subscriber Logical Interfaces Overview on page 321
- Configuring a Pseudowire Subscriber Logical Interface on page 323
- Configuring a Pseudowire Subscriber Logical Interface Device on page 325
- Configuring the Transport Logical Interface for a Pseudowire Subscriber Logical Interface on page 327
- Configuring the Service Logical Interface for a Pseudowire Subscriber Logical Interface on page 330

pseudowire-service (Pseudowire Subscriber Interfaces)

Syntax

```
pseudowire-service {
  device-count number;
}
```

Hierarchy Level [edit chassis]

Release Information Statement introduced in Junos OS Release 13.1.

Description Configure properties for the pseudowire devices on the router.

The remaining statement is explained separately.

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

Related Documentation

- Pseudowire Subscriber Logical Interfaces Overview on page 321
- Configuring a Pseudowire Subscriber Logical Interface on page 323
- Configuring the Maximum Number of Pseudowire Logical Interface Devices Supported on the Router on page 325

push (Dynamic VLANs)

Syntax push;

Hierarchy Level [edit dynamic-profiles *profile-name* interfaces *interface-name* unit *logical-unit-number* input-vlan-map]

Release Information Statement introduced in Junos OS Release 10.4.

Description For dynamic VLAN interfaces, specify the VLAN rewrite operation to add a new VLAN tag to the top of the VLAN stack. An outer VLAN tag is pushed in front of the existing VLAN tag. If you include the **push** statement in the configuration, you must also include the *pop* statement at the **[edit dynamic-profiles *profile-name* interfaces *interface-name* unit *logical-unit-number* output-vlan-map]** hierarchy level.

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

Related Documentation

- *Stacking and Rewriting VLAN Tags for the Layer 2 Wholesale Solution*

qualified-next-hop (Access)

Syntax qualified-next-hop *next-hop*;

Hierarchy Level [edit routing-options access route *ip-prefix</prefix-length>*]

Release Information Statement introduced in Junos OS Release 10.1.
Statement introduced in Junos OS Release 12.3 for ACX Series routers.

Description Configure the qualified next-hop address for an access route.

Options *next-hop*—Specific qualified next-hop address you want to assign to the access route.

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

Related Documentation
- *Examples: Configuring Static Routes*

radius-realm

Syntax radius-realm *radius-realm-string*;

Hierarchy Level [edit interfaces *interface-name* auto-configure vlan-ranges authentication username-include],
[edit interfaces *interface-name* auto-configure stacked-vlan-ranges authentication username-include]

Release Information Statement introduced in Junos OS Release 10.0.

Description Specify that the user-defined RADIUS realm string is appended as a last piece to the username and used by RADIUS to direct the authentication request to a profile that does not allocates addresses.

Options *radius-realm-string*—A string to describe the RADIUS realm.

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

Related Documentation
- Configuring VLAN Interface Username Information for AAA Authentication on page 22

ranges (Dynamic VLAN)

Syntax ranges (any | *low-tag*)-(any | *high-tag*);

Hierarchy Level [edit interfaces *interface-name* auto-configure vlan-ranges dynamic-profile *profile-name*]

Release Information Statement introduced in Junos OS Release 9.5.

Description Configure VLAN ranges for dynamic, auto-sensed VLANs.

Options **any**—The entire VLAN range.

low-tag—The lower limit of the VLAN range.

high-tag—The upper limit of the VLAN range.
Range: 1 through 4094

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

Related Documentation

- *Configuring Single-Level VLAN Ranges for Use with VLAN Dynamic Profiles*

ranges (Dynamic Stacked VLAN)

Syntax ranges (any | *low-tag–high-tag*),(any | *low-tag–high-tag*);

Hierarchy Level [edit interfaces *interface-name* auto-configure stacked-vlan-ranges dynamic-profile *profile-name*]

Release Information Statement introduced in Junos OS Release 9.5.

Description Configure VLAN ranges for dynamic, auto-sensed stacked VLANs.

Options **any**—The entire VLAN range.

low-tag—The lower limit of the VLAN range.

high-tag—The upper limit of the VLAN range.
Range: 1 through 4094

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

Related Documentation

- *Configuring Stacked VLAN Ranges for Use with Stacked VLAN Dynamic Profiles*

route (Access)

Syntax

```
route ip-prefix</prefix-length> {
  metric route-cost;
  next-hop next-hop;
  preference route-distance;
  qualified-next-hop next-hop;
  tag tag-number;
}
```

Hierarchy Level [edit routing-options access]

Release Information Statement introduced in Junos OS Release 10.1.

Description Configure the parameters for access routes.

Options *ip-prefix</prefix-length>*—Specific route prefix that you want to assign to the access route.

The remaining statements are explained separately.

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

Related Documentation

- *Examples: Configuring Static Routes*

routing-instance (PPPoE Service Name Tables)

Syntax routing-instance *routing-instance-name*;

Hierarchy Level [edit protocols pppoe service-name-tables *table-name* service *service-name*],
[edit protocols pppoe service-name-tables *table-name* service *service-name* agent-specifier aci *circuit-id-string* ari *remote-id-string*]

Release Information Statement introduced in Junos OS Release 10.2.

Description Use in conjunction with the **dynamic-profile** statement at the same hierarchy levels to specify the routing instance in which to instantiate a dynamic PPPoE interface. You can associate a routing instance with a named service entry, **empty** service entry, or **any** service entry configured in a PPPoE service name table, or with an agent circuit identifier/agent remote identifier (ACI/ARI) pair defined for these services.

The routing instance associated with a service entry in a PPPoE service name table overrides the routing instance associated with the PPPoE underlying interface on which the dynamic PPPoE interface is created.

If you include the **routing-instance** statement at the **[edit protocols pppoe service-name-tables *table-name* service *service-name* agent-specifier aci *circuit-id-string* ari *remote-id-string*]** hierarchy level, you cannot also include the **static-interface** statement at this level. The **routing-instance** and **static-interface** statements are mutually exclusive for ACI/ARI pair configurations.

Options *routing-instance-name*—Name of the routing instance in which the router instantiates the dynamic PPPoE interface.

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

Related Documentation

- *Configuring PPPoE Service Name Tables*
- Assigning a Dynamic Profile and Routing Instance to a Service Name or ACI/ARI Pair for Dynamic PPPoE Interface Creation on page 150

routing-options

Syntax routing-options { ... }

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

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

Description Configure protocol-independent routing properties.

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

Related Documentation

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

rpf-check (interfaces)

Syntax

```
rpf-check {
  fail-filter filter-name;
  mode loose;
}
```

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

Release Information Statement introduced before Junos OS Release 7.4.

Description Check whether traffic is arriving on an expected path. You can include this statement with the **inet** or **inet6** protocol family only.

The **mode** statement is explained separately.

Options **fail-filter**—A filter to evaluate when packets are received on the interface. If the RPF check fails, this optional filter is evaluated. If the fail filter is not configured, the default action is to silently discard the packet.

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

Related Documentation

- *Configuring Unicast RPF Strict Mode*
- *Configuring Unicast RPF Loose Mode*
- *Example: Configuring Unicast Reverse-Path-Forwarding Check*

rpf-check (Dynamic Profiles)

Syntax

```
rpf-check {
  fail-filter filter-name;
  mode loose;
}
```

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

Release Information Statement introduced in Junos OS Release 9.6.

Description Check whether traffic is arriving on an expected path. You can include this statement with the **inet** protocol family only.

The remaining statements are explained separately.

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

Related Documentation

- *Configuring Unicast RPF*
- *Configuring Unicast RPF and Fail Filters in Dynamic Profiles for Subscriber Interfaces*

schedulers (Class of Service)

Syntax

```
schedulers {
  scheduler-name {
    adjust-minimum rate;
    adjust-percent percentage;
    buffer-size (seconds | percent percentage | remainder | temporal microseconds);
    drop-profile-map loss-priority (any | low | medium-low | medium-high | high) protocol
      (any | non-tcp | tcp) drop-profile profile-name;
    excess-priority [ low | medium-low | medium-high | high | none];
    excess-rate (percent percentage | proportion value);
    priority priority-level;
    shaping-rate (percent percentage | rate);
    transmit-rate (percent percentage | rate | remainder) <exact | rate-limit>;
  }
}
```

Hierarchy Level [edit class-of-service]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 12.1X48 for PTX Series routers.

Description Specify the scheduler name and parameter values.

Options ***scheduler-name***—Name of the scheduler to be configured.

The remaining statements are explained separately.

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

Related Documentation

- *Schedulers Overview*
- *Default Schedulers Overview*
- *Configuring Schedulers*
- *Configuring a Scheduler*

server

Syntax server;

Hierarchy Level [edit interfaces pp0 unit *logical-unit-number* pppoe-options],
[edit logical-systems *logical-system-name* interfaces pp0 unit *logical-unit-number* pppoe-options]

Release Information Statement introduced in Junos OS Release 8.5.

Description Configure the router to operate in the PPPoE server mode. Supported on M120 and M320 Multiservice Edge Routers and MX Series Universal Edge Routers operating as access concentrators.

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

Related Documentation

- *Configuring the PPPoE Server Mode*

server (Dynamic PPPoE)

Syntax server;

Hierarchy Level [edit dynamic-profiles *profile-name* interfaces pp0 unit "$junos-interface-unit" pppoe-options]

Release Information Statement introduced in Junos OS Release 10.1.

Description In a dynamic profile, configure the router to act as a PPPoE server, also known as a remote access concentrator, when a PPPoE logical interface is dynamically created.

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

Related Documentation

- Configuring a Basic PPPoE Dynamic Profile on page 146
- Subscriber Interfaces and PPPoE Overview on page 141

service (Dynamic Service Sets)

Syntax

```
service {
  input {
    service-set service-set-name {
      service-filter filter-name;
    }
    post-service-filter filter-name;
  }
  output {
    service-set service-set-name {
      service-filter filter-name;
    }
  }
}
```

Hierarchy Level

[edit dynamic-profiles *profile-name* interfaces *interface-name* unit *logical-unit-number* family *family*],
[edit dynamic-profiles *profile-name* interfaces pp0 unit "$junos–interface–unit" family *family*]

Release Information

Statement introduced in Junos OS Release 9.5.
Support at the **[edit dynamic-profiles *profile-name* interfaces pp0 unit "$junos-interface-unit" family *family*]** hierarchy level introduced in Junos OS Release 10.1.

Description

Define the service sets and filters to be applied to an interface. This statement is not supported for **family inet6**.

The remaining statements are explained separately.

Required Privilege Level

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

Related Documentation

- *Dynamic Service Sets Overview*
- *Associating Service Sets with Interfaces in a Dynamic Profile*

service-device-pool (L2TP)

Syntax service-device-pool *pool-name*;

Hierarchy Level [edit services l2tp tunnel-group *name*]

Release Information Statement introduced in Junos OS Release 11.4.

Description Assign a pool of service interfaces to the tunnel group to balance traffic across.

NOTE: The service interface configuration is required for static LNS sessions. Either the service interface configuration or the service device pool configuration can be used for dynamic LNS sessions.

Options *pool-name*—Name of the service device pool.

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

Related Documentation

- *Configuring an L2TP Tunnel Group for LNS Sessions with Inline Services Interfaces*

service-filter (Dynamic Service Sets)

Syntax service-filter *filter-name*;

Hierarchy Level [edit dynamic-profiles *profile-name* interfaces *interface-name* unit *logical-unit-number* family *family* service input service-set *service-set-name*],
[edit dynamic-profiles *profile-name* interfaces *interface-name* unit *logical-unit-number* family *family* service output service-set *service-set-name*],
[edit dynamic-profiles *profile-name* interfaces pp0 unit "$junos-interface-unit" family *family* service input service-set *service-set-name*],
[edit dynamic-profiles *profile-name* interfaces pp0 unit "$junos-interface-unit" family *family* service output service-set *service-set-name*]

Release Information Statement introduced in Junos OS Release 9.5.
Support at the **[edit dynamic-profiles *profile-name* interfaces pp0 unit "$junos-interface-unit" family *family* service input service-set *service-set-name*]** and **[edit dynamic-profiles *profile-name* interfaces pp0 unit "$junos-interface-unit" family *family* service output service-set *service-set-name*]** hierarchy levels introduced in Junos OS Release 10.1.

Description Define the filter to be applied to traffic before it is accepted for service processing. Configuration of a service filter is optional; if you include the **service-set** statement without a **service-filter** definition, the router software assumes that the match condition is true and selects the service set for processing automatically. Only the Internet Protocol version 4 (IPv4) protocol family is currently supported for dynamic PPPoE logical interfaces.

Options *filter-name*—Identifies the filter to be applied in service processing.

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

Related Documentation

- *Dynamic Service Sets Overview*
- *Associating Service Sets with Interfaces in a Dynamic Profile*

service-name-table

Syntax

service-name-table *table-name*;

Hierarchy Level

[edit dynamic-profiles *profile-name* interfaces demux0 unit *logical-unit-number* family pppoe],
[edit dynamic-profiles *profile-name* interfaces *interface-name* unit *logical-unit-number* family pppoe],
[edit interfaces *interface-name* unit *logical-unit-number* family pppoe],
[edit interfaces *interface-name* unit *logical-unit-number* pppoe-underlying-options],
[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number* family pppoe],
[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number* pppoe-underlying-options]

Release Information

Statement introduced in Junos OS Release 10.0.
Support at the **[edit ... family pppoe]** hierarchies introduced in Junos OS Release 11.2.

Description

Specify the PPPoE service name table assigned to a PPPoE underlying interface. This underlying interface is configured with either the **encapsulation ppp-over-ether** statement or the **family pppoe** statement; the two statements are mutually exclusive.

NOTE: The **[edit ... family pppoe]** hierarchies are supported only on MX Series routers with MPCs.

Options

table-name—Name of the PPPoE service name table, a string of up to 32 alphanumeric characters.

Required Privilege Level

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

Related Documentation

- *Configuring PPPoE Service Name Tables*
- *Assigning a Service Name Table to a PPPoE Underlying Interface*
- Configuring the PPPoE Family for an Underlying Interface on page 159

service-name-tables

Syntax

```
service-name-tables table-name {
  service service-name {
    drop;
    delay seconds;
    terminate;
    dynamic-profile profile-name;
    routing-instance routing-instance-name;
    max-sessions number;
    agent-specifier {
      aci circuit-id-string ari remote-id-string {
        drop;
        delay seconds;
        terminate;
        dynamic-profile profile-name;
        routing-instance routing-instance-name;
        static-interface interface-name;
      }
    }
  }
}
```

Hierarchy Level [edit protocols pppoe]

Release Information Statement introduced in Junos OS Release 10.0.
dynamic-profile, **routing-instance**, **max-sessions**, and **static-interface** options introduced in Junos OS Release 10.2.

Description Create and configure a PPPoE service name table. Specify the action taken for each service and remote access concentrator on receipt of a PPPoE Active Discovery Initiation (PADI) packet. You can also specify the dynamic profile and routing instance that the router uses to instantiate a dynamic PPPoE interface, and the maximum number of active PPPoE sessions that the router can establish with the specified service. A maximum of 32 PPPoE service name tables is supported per router.

Options ***table-name***—Name of the PPPoE service name table, a string of up to 32 alphanumeric characters.

The remaining statements are explained separately.

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

Related Documentation

- *Configuring PPPoE Service Name Tables*
- *Creating a Service Name Table*

service-set (Dynamic Service Sets)

Syntax

```
service-set service-set-name {
  service-filter filter-name;
}
```

Hierarchy Level

[edit dynamic-profiles *profile-name* interfaces *interface-name* unit *logical-unit-number* family *family* service input],
[edit dynamic-profiles *profile-name* interfaces *interface-name* unit *logical-unit-number* family *family* service output],
[edit dynamic-profiles *profile-name* interfaces pp0 unit "$junos-interface-unit" family *family* service input],
[edit dynamic-profiles *profile-name* interfaces pp0 unit "$junos-interface-unit" family *family* service output]

Release Information

Statement introduced in Junos OS Release 9.5.
Support at the **[edit dynamic-profiles *profile-name* interfaces pp0 unit "$junos-interface-unit" family *family* service input]** and **[edit dynamic-profiles *profile-name* interfaces pp0 unit "$junos-interface-unit" family *family* service output]** hierarchy levels introduced in Junos OS Release 10.1.

Description

Define one or more service sets in a dynamic profile. Service sets are applied to an interface. If you define multiple service sets, the router software evaluates the filters in the order in which they appear in the configuration. Only the Internet Protocol version 4 (IPv4) protocol family is currently supported for dynamic PPPoE logical interfaces.

Options

service-set-name—Name of the service set.

The remaining statement is explained separately.

Required Privilege Level

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

Related Documentation

- *Dynamic Service Sets Overview*
- *Associating Service Sets with Interfaces in a Dynamic Profile*

short-cycle-protection (Static and Dynamic Subscribers)

Syntax

short-cycle-protection <lockout-time-min *minimum-seconds*> <lockout-time-max *maximum-seconds*> <filter [aci]> ;

Hierarchy Level

[edit dynamic-profiles *profile-name* interfaces demux0 unit *logical-unit-number* family pppoe],
[edit dynamic-profiles *profile-name* interfaces *interface-name* unit *logical-unit-number* family pppoe],
[edit dynamic-profiles *profile-name* interfaces *interface-name* unit *logical-unit-number* pppoe-underlying-options],
[edit interfaces demux0 unit *logical-unit-number* family pppoe]
[edit interfaces *interface-name* unit *logical-unit-number* family pppoe],
[edit interfaces *interface-name* unit *logical-unit-number* pppoe-underlying-options],
[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number* family pppoe],
[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number* pppoe-underlying-options]

Release Information

Statement introduced in Junos OS Release 11.4.

Description

Configure the router to temporarily prevent (lock out) a failed or short-lived (also known as short-cycle) PPPoE subscriber session from reconnecting for a default or configurable period of time. You can optionally override the default lockout time, 1 through 300 seconds (5 minutes), by specifying the minimum lockout time and maximum lockout time as part of the **short-cycle-protection** statement. You can optionally specify the lockout based on the ACI, which locks out all PPPoE subscriber sessions that come from the same household and share the same ACI string.

Options

filter aci—(Optional) Agent circuit identifier (ACI) lockout for all subscriber sessions.

lockout-time-min *minimum-seconds*—(Optional) Minimum lockout time for failed or short-lived PPPoE subscriber sessions. The *minimum-seconds* value must be less than or equal to the *maximum-seconds* value. Setting *minimum-seconds* and *maximum-seconds* to the same value causes the lockout time to become fixed at that value.

Range: 1 through 86400 (24 hours)

Default: 1

lockout-time-max *maximum-seconds*—(Optional) Maximum lockout time for failed or short-lived PPPoE subscriber sessions. The *maximum-seconds* value must be equal to or greater than the *minimum-seconds* value. Setting *maximum-seconds* and *minimum-seconds* to the same value causes the lockout time to become fixed at that value.

Range: 1 through 86400 (24 hours)

Default: 300 (5 minutes)

Required Privilege Level

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

Related Documentation

- Configuring Lockout of PPPoE Subscriber Sessions on page 198
- PPPoE Subscriber Session Lockout Overview on page 191
- Understanding the Lockout Period for PPPoE Subscriber Session Lockout on page 196
- Configuring Dynamic PPPoE Subscriber Interfaces Using Dynamic Profiles on page 145
- Example: Configuring a Static PPPoE Subscriber Interface on a Static Underlying VLAN Demux Interface over Aggregated Ethernet on page 165

stacked-vlan-ranges

Syntax

```
stacked-vlan-ranges {
  access-profile profile-name;
  authentication {
    packet-types [packet-types];
    password password-string;
    username-include {
      circuit-type;
      delimiter delimiter-character;
      domain-name domain-name-string;
      interface-name;
      mac-address;
      option-18
      option-37
      option-82;
      radius-realm radius-realm-string;
      user-prefix user-prefix-string;
    }
  }
  dynamic-profile profile-name {
    accept (any | dhcp-v4 | inet);
    ranges (any | low-tag–high-tag),(any | low-tag–high-tag);
  }
  override;
}
```

Hierarchy Level [edit interfaces *interface-name* auto-configure]

Release Information Statement introduced in Junos OS Release 9.5.

Description Configure multiple VLANs. Each VLAN is assigned a VLAN ID number from the range.

The remaining statements are explained separately.

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

Related Documentation

- *Configuring Stacked VLAN Ranges for Use with Stacked VLAN Dynamic Profiles*
- Configuring Interfaces to Support Both Single and Stacked VLANs on page 13

stacked-vlan-tagging

Syntax stacked-vlan-tagging;

Hierarchy Level [edit interfaces *interface-name*]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 12.2 for ACX Series Universal Access Routers.

Description For Gigabit Ethernet IQ interfaces, Gigabit Ethernet, 10-Gigabit Ethernet LAN/WAN PIC, and 100-Gigabit Ethernet Type 5 PIC with CFP, enable stacked VLAN tagging for all logical interfaces on the physical interface.

For pseudowire subscriber interfaces, enable stacked VLAN tagging for logical interfaces on the pseudowire service.

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

Related Documentation

- *Stacking and Rewriting Gigabit Ethernet VLAN Tags Overview*
- *vlan-tags (Stacked VLAN Tags)*

swap (Dynamic VLANs)

Syntax swap;

Hierarchy Level [edit dynamic-profiles *profile-name* interfaces *interface-name* unit *logical-unit-number* input-vlan-map],
[edit dynamic-profiles *profile-name* interfaces *interface-name* unit *logical-unit-number* output-vlan-map]

Release Information Statement introduced in Junos OS Release 10.4.

Description For dynamic VLAN interfaces, specify the VLAN rewrite operation to replace a VLAN tag. The outer VLAN tag of the frame is overwritten with the user-specified VLAN tag information.

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

Related Documentation

- *Rewriting the VLAN Tag on Tagged Frames*
- *Stacking and Rewriting VLAN Tags for the Layer 2 Wholesale Solution*

tag-protocol-id (Dynamic VLANs)

Syntax tag-protocol-id *tpids*;

Hierarchy Level [edit dynamic-profiles *profile-name* interfaces *interface-name* unit *logical-unit-number* input-vlan-map],
[edit dynamic-profiles *profile-name* interfaces *interface-name* unit *logical-unit-number* output-vlan-map]

Release Information Statement introduced in Junos OS Release 10.4.

Description For dynamic VLAN interfaces, configure the outer TPID value. All TPIDs you include in input and output VLAN maps must be among those you specify at the **[edit interfaces *interface-name* gigether-options ethernet-switch-profile tag-protocol-id [*tpids*]]** hierarchy level.

Default If the **tag-protocol-id** statement is not configured, the TPID value is 0x8100.

Options ***tpids***—TPIDs to be accepted on the VLAN. Specify TPIDs in hexadecimal format.

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

Related Documentation
- *Configuring Inner and Outer TPIDs and VLAN IDs*

traffic-control-profiles

Syntax

traffic-control-profiles *profile-name* {
  adjust-minimum *rate*;
  atm-service (cbr | rtvbr | nrtvbr);
  delay-buffer-rate (percent *percentage* | *rate*);
  excess-rate (percent *percentage* | proportion *value*);
  excess-rate-high (percent *percentage* | proportion *value*);
  excess-rate-low (percent *percentage* | proportion *value*);
  guaranteed-rate (percent *percentage* | *rate*) <burst-size *bytes*>;
  max-burst-size *cells*;
  overhead-accounting (frame-mode | cell-mode | frame-mode-bytes | cell-mode-bytes)
    <bytes (*byte-value*)>;
  peak-rate *rate*;
  scheduler-map *map-name*;
  shaping-rate (percent *percentage* | *rate*) <burst-size *bytes*>;
  shaping-rate-excess-high *rate* [burst-size *bytes*];
  shaping-rate-excess-low *rate* [burst-size *bytes*];
  shaping-rate-priority-high *rate* [burst-size *bytes*];
  shaping-rate-priority-low *rate* [burst-size *bytes*];
  shaping-rate-priority-medium *rate* [burst-size *bytes*];
  strict-priority-scheduler;
  sustained-rate *rate*;
}

Hierarchy Level

[edit class-of-service]

Release Information

Statement introduced in Junos OS Release 7.6.

Description

For Gigabit Ethernet IQ, Channelized IQ PICs, FRF.15 and FRF.16 LSQ interfaces, Enhanced Queuing (EQ) DPCs, and PTX Series routers only, configure traffic shaping and scheduling profiles. For Enhanced EQ PICs, EQ DPCs, and PTX Series routers only, you can include the **excess-rate** statement.

Options

profile-name—Name of the traffic-control profile.

The remaining statements are explained separately.

Required Privilege Level

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

Related Documentation

- *Oversubscribing Interface Bandwidth*
- *Understanding Scheduling on PTX Series Routers*
- *output-traffic-control-profile*

traffic-control-profiles (Dynamic CoS Definition)

Syntax

traffic-control-profiles *profile-name* {
  adjust-minimum *rate*;
  delay-buffer-rate (percent *percentage* | *rate*);
  excess-rate (percent *percentage* | proportion *value* | percent $junos-cos-excess-rate);
  excess-rate-high (percent *percentage* | proportion *value*);
  excess-rate-low (percent *percentage* | proportion *value*);
  guaranteed-rate (percent *percentage* | *rate*) <burst-size *bytes*>;
  overhead-accounting (frame-mode | cell-mode) <bytes *byte-value*>;
  scheduler-map *map-name*;
  shaping-rate (percent *percentage* | *rate* | *predefined-variable*) <burst-size *bytes*>;
}

Hierarchy Level

[edit dynamic-profiles *profile-name* class-of-service]

Release Information

Statement introduced in Junos OS Release 9.2.

Description

Configure traffic shaping and scheduling profiles.

Options

profile-name—Name of the traffic-control profile.

The remaining statements are explained separately.

Required Privilege Level

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

Related Documentation

- *Guidelines for Configuring Dynamic CoS for Subscriber Access*
- *Configuring Traffic Scheduling and Shaping for Subscriber Access*
- *Using the CLI to Modify Traffic-Control Profiles That Are Currently Applied to Subscribers*
- output-traffic-control-profile on page 528

underlying-interface

Syntax

underlying-interface *interface-name*;

Hierarchy Level

[edit interfaces pp0 unit *logical-unit-number* pppoe-options],
[edit interfaces demux0 unit *logical-unit-number* demux-options],
[edit logical-systems *logical-system-name* interfaces demux0 unit *logical-unit-number* demux-options],
[edit logical-systems *logical-system-name* interfaces pp0 unit *logical-unit-number* pppoe-options],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* interfaces demux0 unit *logical-unit-number* demux-options],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* interfaces pp0 unit *logical-unit-number* pppoe-options]

Release Information

Statement introduced before Junos OS Release 7.4.
Support for aggregated Ethernet added in Junos OS Release 9.4.

Description

For J Series Services Routers, M120 and M320 Multiservice Edge routers, and MX Series Universal Edge Routers with PPP over Ethernet interfaces, configure the interface on which PPP over Ethernet is running.

For demux interfaces, configure the underlying interface on which the demultiplexing (demux) interface is running.

Options

interface-name—Name of the interface on which PPP over Ethernet or demux is running. For example, **at-0/0/1.0** (ATM VC), **fe-1/0/1.0** (Fast Ethernet interface), **ge-2/0/0.0** (Gigabit Ethernet interface), **ae1.0** (for IP demux on an aggregated Ethernet interface), or **ae1** (for VLAN demux on an aggregated Ethernet interface).

NOTE: Demux interfaces are currently supported on Gigabit Ethernet, Fast Ethernet, 10-Gigabit Ethernet interfaces, or aggregated Ethernet devices.

Required Privilege Level

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

Related Documentation

- *Configuring an IP Demux Underlying Interface*
- *Configuring a VLAN Demux Underlying Interface*
- *Specifying the Demux Underlying Interface*
- *Configuring the PPPoE Underlying Interface*
- *Junos OS Interfaces and Routing Configuration Guide*

underlying-interface (demux0)

Syntax underlying-interface *underlying-interface-name*;

Hierarchy Level [edit dynamic-profiles *profile-name* interfaces demux0 *interface-name* unit unit *logical-unit-number* demux-options]

Release Information Statement introduced in Junos OS Release 9.3.
Support for aggregated Ethernet introduced in Junos OS Release 9.4.

Description Configure the underlying interface on which the demultiplexing (demux) interface is running.

Options ***underlying-interface-name***—Either the specific name of the interface on which the DHCP discover packet arrives or one of the following interface variables:

- **$junos-underlying-interface** when configuring dynamic IP demux interfaces.
- **$junos-interface-ifd-name** when configuring dynamic VLAN demux interfaces.

The variable is used to specify the underlying interface when a new demux interface is dynamically created. The variable is dynamically replaced with the underlying interface that DHCP supplies when the subscriber logs in.

NOTE: Logical demux interfaces are currently supported on Gigabit Ethernet, Fast Ethernet, 10-Gigabit Ethernet, or aggregated Ethernet interfaces.

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

Related Documentation

- *Configuring Static Subscriber Interfaces Using IP Demux Interfaces*
- Configuring Dynamic Subscriber Interfaces Using IP Demux Interfaces in Dynamic Profiles on page 77
- *Configuring Static Subscriber Interfaces Using VLAN Demux Interfaces*
- Configuring Dynamic Subscriber Interfaces Using VLAN Demux Interfaces in Dynamic Profiles on page 79
- Dynamic PPPoE Subscriber Interfaces over Static Underlying Interfaces Overview on page 153

underlying-interface (Dynamic PPPoE)

Syntax underlying-interface *interface-name*;

Hierarchy Level [edit dynamic-profiles *profile-name* interfaces pp0 unit "$junos-interface-unit" pppoe-options]

Release Information Statement introduced in Junos OS Release 10.1.

Description In a dynamic profile, configure the underlying interface on which the router creates the dynamic PPPoE logical interface.

Options *interface-name*—Variable used to specify the name of the underlying interface on which the PPPoE logical interface is dynamically created. In the **underlying-interface *interface-name*** statement for dynamic PPPoE logical interfaces, you must use the predefined variable **$junos-underlying-interface** in place of ***interface-name***. When the router creates the dynamic PPPoE interface, the **$junos-underlying-interface** predefined variable is dynamically replaced with the name of the underlying interface supplied by the network when the subscriber logs in.

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

Related Documentation

- Configuring a Basic PPPoE Dynamic Profile on page 146
- Dynamic PPPoE Subscriber Interfaces over Static Underlying Interfaces Overview on page 153

unit

Syntax

```
unit logical-unit-number {
  accept-source-mac {
    mac-address mac-address {
      policer {
        input cos-policer-name;
        output cos-policer-name;
      }
    }
  }
  accounting-profile name;
  advisory-options {
    downstream-rate rate;
    upstream-rate rate;
  }
  allow-any-vci;
  atm-scheduler-map (map-name | default);
  backup-options {
    interface interface-name;
  }
  bandwidth rate;
  cell-bundle-size cells;
  clear-dont-fragment-bit;
  compression {
    rtp {
      maximum-contexts number <force>;
      f-max-period number;
      queues [queue-numbers];
      port {
        minimum port-number;
        maximum port-number;
      }
    }
  }
  compression-device interface-name;
  copy-tos-to-outer-ip-header;
  demux-destination family;
  demux-source family;
  demux-options {
    underlying-interface interface-name;
  }
  description text;
  interface {
    l2tp-interface-id name;
    (dedicated | shared);
  }
  dialer-options {
    activation-delay seconds;
    callback;
    callback-wait-period time;
    deactivation-delay seconds;
    dial-string [dial-string-numbers];
    idle-timeout seconds;
```



```
   incoming-map {
      caller caller-id) | accept-all;
      initial-route-check seconds;
      load-interval seconds;
      load-threshold percent;
      pool pool-name;
      redial-delay time;
      watch-list {
         [routes];
      }
   }
}
disable;
disable-mlppp-inner-ppp-pfc;
dlci dlci-identifier;
drop-timeout milliseconds;
dynamic-call-admission-control {
   activation-priority priority;
   bearer-bandwidth-limit kilobits-per-second;
}
encapsulation type;
epd-threshold cells plp1 cells;
family family-name {
   ... the family subhierarchy appears after the main [edit interfaces interface-name unit
      logical-unit-number] hierarchy ...
}
fragment-threshold bytes;
inner-vlan-id-range start start-id end end-id;
input-vlan-map {
   (pop | pop-pop | pop-swap | push | push-push | swap |
   swap-push | swap-swap);
   inner-tag-protocol-id tpid;
   inner-vlan-id number;
   tag-protocol-id tpid;
   vlan-id number;
}
interleave-fragments;
inverse-arp;
layer2-policer {
   input-policer policer-name;
   input-three-color policer-name;
   output-policer policer-name;
   output-three-color policer-name;
}
link-layer-overhead percent;
minimum-links number;
mrru bytes;
multicast-dlci dlci-identifier;
multicast-vci vpi-identifier.vci-identifier;
multilink-max-classes number;
multipoint;
oam-liveness {
   up-count cells;
   down-count cells;
}
oam-period (disable | seconds);
```

```
output-vlan-map {
  (pop | pop-pop | pop-swap | push | push-push | swap |
  swap-push | swap-swap);
  inner-tag-protocol-id tpid;
  inner-vlan-id number;
  tag-protocol-id tpid;
  vlan-id number;
}
passive-monitor-mode;
peer-unit unit-number;
plp-to-clp;
point-to-point;
ppp-options {
  chap {
    access-profile name;
    default-chap-secret name;
    local-name name;
    passive;
  }
  compression {
    acfc;
    pfc;
  }
  dynamic-profile profile-name;
  lcp-restart-timer milliseconds;
  loopback-clear-timer seconds;
  ncp-restart-timer milliseconds;
  pap {
    access-profile name;
    default-pap-password password;
    local-name name;
    local-password password;
    passive;
  }
}
pppoe-options {
  access-concentrator name;
  auto-reconnect seconds;
  (client | server);
  service-name name;
  underlying-interface interface-name;
}
pppoe-underlying-options {
  access-concentrator name;
  direct-connect;
  dynamic-profile profile-name;
  max-sessions number;
}
proxy-arp;
service-domain (inside | outside);
shaping {
  (cbr rate | rtvbr peak rate sustained rate burst length | vbr peak rate sustained rate burst
    length);
  queue-length number;
}
short-sequence;
```



```
targeted-distribution;
transmit-weight number;
(traps | no-traps);
trunk-bandwidth rate;
trunk-id number;
tunnel {
  backup-destination address;
  destination address;
  key number;
  routing-instance {
    destination routing-instance-name;
  }
  source source-address;
  ttl number;
}
vci vpi-identifier.vci-identifier;
vci-range start start-vci end end-vci;
vpi vpi-identifier;
vlan-id number;
vlan-id-range number-number;
vlan-tags inner tpid.vlan-id outer tpid.vlan-id;
family family {
  accounting {
    destination-class-usage;
    source-class-usage {
      (input | output | input output);
    }
  }
  access-concentrator name;
  address address {
    ... the address subhierarchy appears after the main [edit interfaces interface-name unit
      logical-unit-number family family-name] hierarchy ...
  }
  bundle interface-name;
  core-facing;
  demux-destination {
    destination-prefix;
  }
  demux-source {
    source-prefix;
  }
  direct-connect;
  duplicate-protection;
  dynamic-profile profile-name;
  filter {
    group filter-group-number;
    input filter-name;
    input-list [filter-names];
    output filter-name;
    output-list [filter-names];
  }
  interface-mode (access | trunk);
  ipsec-sa sa-name;
  keep-address-and-control;
  mac-validate (loose | strict);
  max-sessions number;
```

```
mtu bytes;
multicast-only;
no-redirects;
policer {
   arp policer-template-name;
   input policer-template-name;
   output policer-template-name;
}
primary;
protocols [inet iso mpls];
proxy inet-address address;
receive-options-packets;
receive-ttl-exceeded;
remote (inet-address address | mac-address address);
rpf-check {
   fail-filter filter-name
   mode loose;
}
sampling {
   input;
   output;
}
service {
   input {
      post-service-filter filter-name;
      service-set service-set-name <service-filter filter-name>;
   }
   output {
      service-set service-set-name <service-filter filter-name>;
   }
}
service-name-table table-name
(translate-discard-eligible | no-translate-discard-eligible);
(translate-fecn-and-becn | no-translate-fecn-and-becn);
translate-plp-control-word-de;
unnumbered-address interface-name destination address destination-profile profile-name;
vlan-id number;
vlan-id-list [number number-number];
address address {
   arp ip-address (mac | multicast-mac) mac-address <publish>;
   broadcast address;
   destination address;
   destination-profile name;
   eui-64;
   master-only;
   multipoint-destination address {
      dlci dlci-identifier;
      epd-threshold cells <plp1 cells>;
      inverse-arp;
      oam-liveness {
         up-count cells;
         down-count cells;
      }
      oam-period (disable | seconds);
      shaping {
```



```
            (cbr rate | rtvbr burst length peak rate sustained rate | vbr burst length peak rate
              sustained rate);
            queue-length number;
          }
          vci vpi-identifier.vci-identifier;
        }
        preferred;
        primary;
        (vrrp-group | vrrp-inet6-group) group-number {
          (accept-data | no-accept-data);
          advertise–interval seconds;
          authentication-type authentication;
          authentication-key key;
          fast-interval milliseconds;
          (preempt | no-preempt) {
            hold-time seconds;
          }
          priority number;
          track {
            interface interface-name {
              bandwidth-threshold bits-per-second priority-cost number;
            }
            priority-hold-time seconds;
            route ip-address/prefix-length routing-instance instance-name priority-cost cost;
          }
          virtual-address [addresses];
          virtual-link-local-address ipv6–address;
          vrrp-inherit-from {
            active-interface interface-name;
            active-group group-number;
          }
        }
      }
    }
  }
```

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

Release Information Statement introduced before Junos OS Release 7.4.

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

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

Range: 0 through 1,073,741,823 for demux and PPPoE static interfaces. 0 through 16,385 for all other static interface types.

The remaining statements are explained separately.

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

unit (Dynamic Demux Interface)

Syntax

```
unit logical-unit-number {
  demux-options {
    underlying-interface interface-name
  }
  family family {
    access-concentrator name;
    address address;
    demux-source {
      source-address;
    }
    direct-connect;
    duplicate-protection;
    dynamic-profile profile-name;
    filter {
      input filter-name;
      output filter-name;
    }
    mac-validate (loose | strict):
    max-sessions number;
    max-sessions-vsa-ignore;
    rpf-check {
      fail-filter filter-name;
      mode loose;
    }
    service-name-table table-name
    short-cycle-protection <lockout-time-min minimum-seconds lockout-time-max
      maximum-seconds>;
    unnumbered-address interface-name <preferred-source-address address>;
  }
  filter {
    input filter-name;
    output filter-name;
  }
}
vlan-id number;
```

Hierarchy Level [edit dynamic-profiles *profile-name* interfaces demux0]

Release Information Statement introduced in Junos OS Release 9.3.

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

Options *logical-unit-number*—Either the specific unit number of the interface or the unit number variable (**$junos-interface-unit**). The variable is used to specify the unit of the interface when a new demux interface is dynamically created. The static unit number variable is dynamically replaced with the unit number that DHCP supplies when the subscriber logs in.

The remaining statements are explained separately.

Required Privilege Level

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

Related Documentation

- Configuring Dynamic Subscriber Interfaces Using IP Demux Interfaces in Dynamic Profiles on page 77
- *Configuring an IP Demultiplexing Interface*

unit (Dynamic Interface Sets)

Syntax

```
unit logical-unit-number {
  advisory-options {
    downstream-rate rate;
    upstream-rate rate;
  }
}
```

Hierarchy Level

[edit dynamic-profiles *profile-name* interfaces interface-set *interface-set-name* interface *interface-name*]

Release Information

Statement introduced in Junos OS Release 10.4.

Description

Apply the logical interface unit to the interface set.

Options

logical-unit-number—One of the following options:

- **$junos-underlying-interface-unit**—For static VLANs, the unit number variable. The static unit number variable is dynamically replaced with the client unit number when the client session begins. The client unit number is specified by the DHCP when it accesses the subscriber network.
- **$junos-interface-unit**—For dynamic demux and dynamic PPPoE interfaces, the unit number variable. The static unit number variable is dynamically replaced with the client unit number when the client session begins. The client unit number is specified by the DHCP or PPP when it accesses the subscriber network.
- *value*—Specific unit number of the interface you want to assign to the dynamic-profile

Range: 0 through 1,073,741,823.

The remaining statements are explained separately.

Required Privilege Level

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

Related Documentation

- Configuring Dynamic VLAN Subscriber Interfaces Based on Agent Circuit Identifier Information on page 36
- *Applying Traffic Shaping and Scheduling to a Subscriber Interface in a Dynamic Profile*
- *Configuring an Interface Set of Subscribers in a Dynamic Profile*
- Agent Circuit Identifier-Based Dynamic VLANs Components Overview on page 27
- *Guidelines for Configuring Dynamic CoS for Subscriber Access*

unit (Dynamic PPPoE)

Syntax

```
unit logical-unit-number {
  keepalives interval seconds;
  no-keepalives;
  pppoe-options {
    underlying-interface interface-name;
    server;
  }
  ppp-options {
    authentication [ authentication-protocols ];
    chap {
      challenge-length minimum minimum-length maximum maximum-length;
    }
    pap;
  }
  family inet {
    unnumbered-address interface-name;
    address address;
    service {
      input {
        service-set service-set-name {
          service-filter filter-name;
        }
        post-service-filter filter-name;
      }
      output {
        service-set service-set-name {
          service-filter filter-name;
        }
      }
    }
    filter {
      input filter-name {
        precedence precedence;
      }
      output filter-name {
        precedence precedence;
      }
    }
  }
  filter {
    input filter-name;
    output filter-name;
  }
}
```

Hierarchy Level [edit dynamic-profiles *profile-name* interfaces pp0]

Release Information Statement introduced in Junos OS Release 10.1.

Description In a dynamic profile, configure a logical unit number for the dynamic PPPoE logical interface. You must configure a logical interface to be able to use the router.

Options

logical-unit-number—Variable used to specify the unit number when the PPPoE logical interface is dynamically created. In the **unit** ***logical-unit-number*** statement for dynamic PPPoE logical interfaces, you must use the predefined variable **$junos-interface-unit** in place of ***logical-unit-number***. The **$junos-interface-unit** predefined variable is dynamically replaced with the unit number supplied by the router when the subscriber logs in.

The remaining statements are explained separately.

Required Privilege Level

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

Related Documentation

- Configuring a Basic PPPoE Dynamic Profile on page 146
- Dynamic PPPoE Subscriber Interfaces over Static Underlying Interfaces Overview on page 153

unit (Dynamic Profiles Standard Interface)

Syntax

```
unit logical-unit-number {
  auto-configure {
    agent-circuit-identifier {
      dynamic-profile profile-name;
    }
  }
  dial-options {
    ipsec-interface-id name;
    l2tp-interface-id name;
    (shared | dedicated);
  }
  encapsulation (atm-ccc-cell-relay | atm-ccc-vc-mux | atm-cisco-nlpid | atm-tcc-vc-mux
    | atm-mlppp-llc | atm-nlpid | atm-ppp-llc | atm-ppp-vc-mux | atm-snap | atm-tcc-snap
    | atm-vc-mux | ether-over-atm-llc | ether-vpls-over-atm-llc | ether-vpls-over-fr |
    ether-vpls-over-ppp | ethernet | frame-relay-ccc | frame-relay-ppp | frame-relay-tcc |
    frame-relay-ether-type | frame-relay-ether-type-tcc | multilink-frame-relay-end-to-end
    | multilink-ppp | ppp-over-ether | ppp-over-ether-over-atm-llc | vlan-bridge | vlan-ccc |
    vlan-vci-ccc | vlan-tcc | vlan-vpls);
  family family {
    access-concentrator name;
    address address;
    direct-connect;
    duplicate-protection;
    dynamic-profile profile-name;
    filter {
      adf {
        counter;
        input-precedence precedence;
        not-mandatory;
        output-precedence precedence;
        rule rule-value;
      }
      input filter-name (
        precedence precedence;
      }
      output filter-name {
        precedence precedence;
      }
    }
    max-sessions number;
    max-sessions-vsa-ignore;
    rpf-check {
      fail-filter filter-name;
      mode loose;
    }
    service {
      input {
        service-set service-set-name {
          service-filter filter-name;
        }
        post-service-filter filter-name;
      }
```

```
                input-vlan-map {
                    inner-tag-protocol-id tpid;
                    inner-vlan-id number;
                    (push | swap);
                    tag-protocol-id tpid;
                    vlan-id number;
                }
                output {
                    service-set service-set-name {
                        service-filter filter-name;
                    }
                }
                output-vlan-map {
                    inner-tag-protocol-id tpid;
                    inner-vlan-id number;
                    (pop | swap);
                    tag-protocol-id tpid;
                    vlan-id number;
                }
            }
            service-name-table table-name
            short-cycle-protection <lockout-time-min minimum-seconds lockout-time-max
                maximum-seconds>;
            unnumbered-address interface-name <preferred-source-address address>;
            filter {
                input filter-name;
                output filter-name;
            }
            keepalives {
                interval seconds;
                }
            ppp-options {
                chap;
                pap;
            }
            vlan-id number;
            vlan-tags outer [tpid].vlan-id [inner [tpid].vlan-id];
        }
    }
```

Hierarchy Level [edit dynamic-profiles *profile-name* interfaces *interface-name*]

Release Information Statement introduced in Junos OS Release 9.2.

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

Options *logical-unit-number*—The specific unit number of the interface you want to assign to the dynamic profile, or one of the following Junos OS predefined variables:

- **$junos-underlying-interface-unit**—For static VLANs, the unit number variable. The static unit number variable is dynamically replaced with the client unit number when the client session begins. The client unit number is specified by the DHCP when it accesses the subscriber network.
- **$junos-interface-unit**—The unit number variable on a dynamic underlying VLAN interface for which you want to enable the creation of dynamic VLAN subscriber interfaces based on agent circuit identifier information.

The remaining statements are explained separately.

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

Related Documentation

- Configuring Dynamic Underlying VLAN Interfaces to Use Agent Circuit Identifier Information on page 33
- Configuring Static Underlying VLAN Interfaces to Use Agent Circuit Identifier Information on page 35
- Agent Circuit Identifier-Based Dynamic VLANs Components Overview on page 27

unnumbered-address (Dynamic PPPoE)

Syntax unnumbered-address *interface-name*;

Hierarchy Level [edit dynamic-profiles *profile-name* interfaces pp0 unit "$junos-interface-unit" family inet]

Release Information Statement introduced in Junos OS Release 10.1.

Description For dynamic PPPoE interfaces, enable the local address to be derived from the specified interface. Configuring unnumbered Ethernet interfaces enables IP processing on the interface without assigning an explicit IP address to the interface.

Options ***interface-name***—Interface from which the local address is derived. The interface name must include a logical unit number and must have a configured address.

The **destination** statement is explained separately.

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

Related Documentation

- Configuring a Basic PPPoE Dynamic Profile on page 146
- Dynamic PPPoE Subscriber Interfaces over Static Underlying Interfaces Overview on page 153

unnumbered-address (Dynamic Profiles)

Syntax unnumbered-address *interface-name* <preferred-source-address *address*>;

Hierarchy Level [edit dynamic-profiles *profile-name* interfaces *interface-name* unit *logical-unit-number* family *family*],
[edit dynamic-profiles *profile-name* interfaces demux0 unit *logical-unit-number* family *family*]

Release Information Statement introduced in Junos OS Release 9.2.
$junos-preferred-source-address variable support added in Junos OS Release 9.6.
Support for the **$junos-loopback-interface** predefined variable introduced in Junos OS Release 9.6.

Description For Ethernet interfaces, enable the local address to be derived from the specified interface. Configuring unnumbered Ethernet interfaces enables IP processing on the interface without assigning an explicit IP address to the interface. To configure unnumbered address dynamically, include the **$junos-loopback-interface-address** predefined variable.

You can configure unnumbered address support on Ethernet interfaces for IPv4 and IPv6 address families.

Options *interface-name*—Name of the interface from which the local address is derived. The specified interface must have a logical unit number, a configured IP address, and must not be an unnumbered interface. This value can be a specific interface name or the **$junos-loopback-interface** dynamic variable.

When defining the **unnumbered-address** statement using a static interface, keep the following in mind:

- If you choose to include the **routing-instance** statement at the **[edit dynamic-profiles]** hierarchy level, that statement must be configured with a valid, static routing instance value. In addition, whatever static unnumbered interface you specify must belong to that routing instance.
- If you choose to not include the **routing-instance** statement at the **[edit dynamic-profiles]** hierarchy level, the unnumbered-address statement uses the default routing instance. The use of the default routing instance requires that the unnumbered interface be configured statically and that it reside in the default routing instance.

When defining the **unnumbered-address** statement using the **$junos-loopback-interface** dynamic variable, keep the following in mind:

- To use the **$junos-loopback-interface** dynamic variable, the dynamic profile must also contain the **routing-instance** statement configured with the **$junos-routing-instance** dynamic variable at the [edit dynamic-profiles] hierarchy level.
- The applied loopback interface is based on the dynamically obtained routing instance of the subscriber.

address—(Optional) Secondary IP address of the donor interface. Configuring the preferred source address enables you to use an IP address other than the primary IP address on some of the unnumbered Ethernet interfaces in your network. This value can be a static IP address, the **$junos-preferred-source-address** dynamic variable for the inet family, or **$junos-preferred-source-ipv6-address** dynamic variable for the inet6 family.

When defining the **preferred-source-address** value using a static IP address, keep the following in mind:

- The unnumbered interface must be statically configured.
- The IP address specified as the **preferred-source-address** must be configured in the specified unnumbered interface.

When defining the **preferred-source-address** value using the **$junos-preferred-source-address** or **$junos-preferred-source-ipv6-address** dynamic variables, keep the following in mind:

- You must configure the **unnumbered-address** statement using the **$junos-loopback-interface** dynamic variable.
- You must configure the **routing-instance** statement using the **$junos-routing-instance** dynamic variable at the **[edit dynamic-profiles]** hierarchy level.
- The preferred source address chosen is based on the dynamically applied loopback address which is in turn derived from the dynamically obtained routing instance of the subscriber. The configured loopback address with the closest network match to the user IP address is selected as the preferred source address.

Required Privilege Level

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

Related Documentation

- *Dynamic Profiles Overview*

unnumbered-address (PPP)

Syntax unnumbered-address *interface-name* destination *address* destination-profile *profile-name*;

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

Release Information Statement introduced before Junos OS Release 7.4.

Description For interfaces with PPP encapsulation, enable the local address to be derived from the specified interface.

Options ***interface-name***—Interface from which the local address is derived. The interface name must include a logical unit number and must have a configured address.

The remaining statements are explained separately.

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

Related Documentation

- *Configuring IPCP Options*
- *address*
- *negotiate-address*
- *Junos OS Administration Library for Routing Devices*

untagged

Syntax untagged;

Hierarchy Level [edit interfaces ps0]

Release Information Statement introduced in Junos OS Release 13.1.

Description Specify that the router supports untagged traffic on pseudowire subscriber interfaces.

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

Related Documentation

- Configuring a Pseudowire Subscriber Logical Interface Device on page 325

use-primary (DHCP Local Server)

Syntax use-primary *primary-profile-name*;

Hierarchy Level [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* system services dhcp-local-server dynamic-profile *profile-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* system services dhcp-local-server group *group-name* dynamic-profile *profile-name*],
[edit logical-systems *logical-system-name* system services dhcp-local-server dynamic-profile *profile-name*],
[edit logical-systems *logical-system-name* system services dhcp-local-server group *group-name* dynamic-profile *profile-name*],
[edit routing-instances *routing-instance-name* system services dhcp-local-server dynamic-profile *profile-name*],
[edit routing-instances *routing-instance-name* system services dhcp-local-server group *group-name* dynamic-profile *profile-name*],
[edit system services dhcp-local-server dynamic-profile *profile-name*],
[edit system services dhcp-local-server group *group-name* dynamic-profile *profile-name*]

Release Information Statement introduced in Junos OS Release 9.3.
Statement introduced in Junos OS Release 12.3R2 for EX Series switches.

Description Specify the dynamic profile to configure as the primary dynamic profile. The primary dynamic profile is instantiated when the first subscriber or DHCP client logs in. Subsequent subscribers (or clients) are not assigned the primary dynamic profile; instead, they are assigned the dynamic profile specified for the interface. When the first subscriber (or client) logs out, the next subscriber (or client) that logs in is assigned the primary dynamic profile.

Options ***primary-profile-name***—Name of the dynamic profile to configure as the primary dynamic profile

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

Related Documentation

- Attaching Dynamic Profiles to DHCP Subscriber Interfaces or DHCP Client Interfaces on page 114

username-include

Syntax

```
username-include {
  circuit-type;
  delimiter delimiter-character;
  domain-name domain-name-string;
  interface-name;
  mac-address;
  option-18;
  option-37;
  option-82 <circuit-id> <remote-id>;
  radius-realm radius-realm-string;
  user-prefix user-prefix-string;
}
```

Hierarchy Level

[edit interfaces *interface-name* auto-configure vlan-ranges authentication],
[edit interfaces *interface-name* auto-configure stacked-vlan-ranges authentication]

Release Information

Statement introduced in Junos OS Release 10.0.

Description

Configure the username that the router passes to the external AAA server. You must include at least one of the optional statements for the username to be valid. If you do not configure a username, the router accesses the local authentication service only and does not use external authentication services, such as RADIUS.

The username takes the format ***user-prefix mac-address circuit-type option–82 interface-name domain-name radius-realm*** with each component separated by whatever delimiter you choose.

The remaining statements are explained separately.

Required Privilege Level

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

Related Documentation

- Configuring VLAN Interface Username Information for AAA Authentication on page 22
- Using DHCP Option 82 Suboptions in Authentication Usernames for Autosense VLANs on page 23
- Using DHCP Option 18 and Option 37 in Authentication Usernames for DHCPv6 Autosense VLANs on page 24

user-prefix

Syntax user-prefix *user-prefix-string*;

Hierarchy Level [edit interfaces *interface-name* auto-configure vlan-ranges authentication username-include],
[edit interfaces *interface-name* auto-configure stacked-vlan-ranges authentication username-include]

Release Information Statement introduced in Junos OS Release 10.0.

Description Specify the user prefix that is concatenated with the username during the subscriber authentication process.

Options ***user-prefix-string***—The user prefix string.

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

Related Documentation

- Configuring VLAN Interface Username Information for AAA Authentication on page 22

vci

Syntax

vci *vpi-identifier*.*vci-identifier*;

Hierarchy Level

[edit interfaces at-*fpc/pic/port* unit *logical-unit-number*],
[edit interfaces at-*fpc/pic/port* unit *logical-unit-number* family *family* address *address* multipoint-destination *address*],
[edit logical-systems *logical-system-name* interfaces at-*fpc/pic/port* unit *logical-unit-number*],
[edit logical-systems *logical-system-name* interfaces at-*fpc/pic/port* unit *logical-unit-number* family *family* address *address* multipoint-destination *address*]

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 11.1 for the QFX Series.
Statement introduced in Junos OS Release 12.2 for the ACX Series Universal Access routers.

Description

For ATM point-to-point logical interfaces only, configure the virtual circuit identifier (VCI) and virtual path identifier (VPI).

To configure a VPI for a point-to-multipoint interface, specify the VPI in the *multipoint-destination* statement.

VCIs 0 through 31 are reserved for specific ATM values designated by the ATM Forum.

Options

vci-identifier—ATM virtual circuit identifier. Unless you configure the interface to use promiscuous mode, this value cannot exceed the highest-numbered VC configured for the interface with the **maximum-vcs** option of the **vpi** statement.

Range: 0 through 4089 or 0 through 65,535 with promiscuous mode, with VCIs 0 through 31 reserved.

vpi-identifier—ATM virtual path identifier.

Range: 0 through 255

Default: 0

Required Privilege Level

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

Related Documentation

- *Configuring a Point-to-Point ATM1 or ATM2 IQ Connection*
- *Applying Scheduler Maps to Logical ATM Interfaces*
- *multipoint-destination*
- *promiscuous-mode*
- *vpi (ATM CCC Cell-Relay Promiscuous Mode)*

vlan-id (Dynamic Profiles)

Syntax vlan-id (*number* | none);

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

Release Information Statement introduced in Junos OS Release 9.5.
VLAN demux interface support introduced in Junos OS Release 10.2.

Description For VLAN demux, Fast Ethernet, Gigabit Ethernet, and Aggregated Ethernet interfaces only, bind a 802.1Q VLAN tag ID to a logical interface.

Options *number*—A valid VLAN identifier. When used in the **dynamic-profiles** hierarchy, specify the **$junos-vlan-id** predefined variable to dynamically obtain the VLAN identifier.

none—Enable the use of untagged pseudo-wire frames on dynamic interfaces.

- For aggregated Ethernet, 4-port, 8-port, and 12-port Fast Ethernet PICs, and for management and internal Ethernet interfaces, 1 through 1023.
- For 48-port Fast Ethernet and Gigabit Ethernet PICs, 1 through 4094.
- VLAN ID 0 is reserved for tagging the priority of frames.

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

Related Documentation

- *Configuring Static Subscriber Interfaces Using VLAN Demux Interfaces*
- Configuring Dynamic Subscriber Interfaces Using VLAN Demux Interfaces in Dynamic Profiles on page 79

vlan-id (Dynamic VLANs)

Syntax vlan-id *number*;

Hierarchy Level [edit dynamic-profiles *profile-name* interfaces *interface-name* unit *logical-unit-number* input-vlan-map],
[edit dynamic-profiles *profile-name* interfaces *interface-name* unit *logical-unit-number* output-vlan-map]

Release Information Statement introduced in Junos OS Release 10.4.

Description For dynamic VLAN interfaces, specify the line VLAN identifiers to be rewritten at the input or output interface.

You cannot include the **vlan-id** statement with the **swap** statement, **swap-push** statement, **push-push** statement, or **push-swap** statement at the **[edit** dynamic-profiles ***profile-name*** interfaces ***interface-name*** unit ***logical-unit-number*** output-vlan-map**]** hierarchy level. If you include any of those statements in the output VLAN map, the VLAN ID in the outgoing frame is rewritten to the **vlan-id** statement that you include at the **[edit** dynamic-profiles ***profile-name*** interfaces ***interface-name*** unit ***logical-unit-number*****]** hierarchy level.

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

Related Documentation
- *Rewriting the VLAN Tag on Tagged Frames*
- *Binding VLAN IDs to Logical Interfaces*

vlan-ranges

Syntax

```
vlan-ranges {
  access-profile profile-name;
  authentication {
    packet-types [packet-types];
    password password-string;
    username-include {
      circuit-type;
      delimiter delimiter-character;
      domain-name domain-name-string;
      interface-name;
      mac-address;
      option-82 <circuit-id> <remote-id>;
      radius-realm radius-realm-string;
      user-prefix user-prefix-string;
    }
  }
  dynamic-profile profile-name {
    accept (any | dhcp-v4 | inet);
    ranges (any | low-tag)–(any | high-tag);
  }
  override;
}
```

Hierarchy Level [edit interfaces *interface-name* auto-configure]

Release Information Statement introduced in Junos OS Release 9.5.

Description Configure multiple VLANs. Each VLAN is assigned a VLAN ID number from the range.

The remaining statements are explained separately.

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

Related Documentation

- *Configuring Single-Level VLAN Ranges for Use with VLAN Dynamic Profiles*
- Configuring Interfaces to Support Both Single and Stacked VLANs on page 13

vlan-tagging

Syntax vlan-tagging;

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

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.
Statement introduced in Junos OS Release 12.2 for ACX Series Universal Access Routers.
Statement introduced in Junos OS Release 13.2 for PTX Series Routers.

Description For Fast Ethernet and Gigabit Ethernet interfaces, aggregated Ethernet interfaces configured for VPLS, and pseudowire subscriber interfaces, enable the reception and transmission of 802.1Q VLAN-tagged frames on the interface.

NOTE: On EX Series switches except for EX4300 and EX9200 switches, the **vlan-tagging** and **family ethernet-switching** statements cannot be configured on the same interface. Interfaces on EX2200, EX3200, EX3300, EX4200, and EX4500 switches are set to **family ethernet-switching** by the default factory configuration. EX6200 and EX8200 switch interfaces do not have a default **family** setting.

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

Related Documentation

- *Example: Configuring Layer 3 Subinterfaces for a Distribution Switch and an Access Switch*
- *Example: Configuring BGP Autodiscovery for LDP VPLS*
- *Configuring a Layer 3 Subinterface (CLI Procedure)*
- *Configuring Tagged Aggregated Ethernet Interfaces*
- *Configuring Interfaces for VPLS Routing*
- *Enabling VLAN Tagging*
- *802.1Q VLANs Overview*
- *vlan-id*

vlan-tagging

Syntax vlan-tagging;

Hierarchy Level [edit dynamic-profiles *profile-name* interfaces *interface-name*],
[edit interfaces *interface-name*]

Release Information Statement introduced in Junos OS Release 9.2.

Description For Fast Ethernet and Gigabit Ethernet interfaces and aggregated Ethernet interfaces configured for VPLS, enable the reception and transmission of 802.1Q VLAN-tagged frames on the interface.

NOTE: For Ethernet, Fast Ethernet, Tri-Rate Ethernet copper, Gigabit Ethernet, 10-Gigabit Ethernet, and aggregated Ethernet interfaces supporting VPLS, the Junos OS supports a subset of the IEEE 802.1Q standard for channelizing an Ethernet interface into multiple logical interfaces, allowing many hosts to be connected to the same Gigabit Ethernet switch, but preventing them from being in the same routing or bridging domain.

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

Related Documentation

- *Configuring VLAN Ranges for Use with Dynamic Profiles*
- *Configuring Static Subscriber Interfaces in Dynamic Profiles*
- *Configuring the L2TP LNS Peer Interface*

vlan-tags

Syntax vlan-tags outer [*tpid*].*vlan-id* [inner [*tpid*].*vlan-id*];

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

Release Information Statement introduced in Junos OS Release 9.5.
VLAN demux interface support introduced in Junos OS Release 10.2.

Description For Gigabit Ethernet IQ and IQE interfaces only, binds TPIDs and 802.1Q VLAN tag IDs to a logical interface. You must include the **stacked-vlan-tagging** statement at the **[edit interfaces *interface-name*]** hierarchy level.

NOTE: The **inner-range *vid1–vid2*** option is supported on MX Series routers with IQE PICs only.

Options **inner [*tpid*].*vlan-id***—A TPID (optional) and a valid VLAN identifier in the format *tpid.vlan-id*. When used in the **dynamic-profiles** hierarchy, specify the **$junos-vlan-id** predefined variable to dynamically obtain the VLAN ID.

Range: For VLAN ID, 1 through 4094. VLAN ID 0 is reserved for tagging the priority of frames.

outer [*tpid*].*vlan-id*—A TPID (optional) and a valid VLAN identifier in the format *tpid.vlan-id*. When used in the **dynamic-profiles** hierarchy, specify the **$junos-stacked-vlan-id** predefined variable.

Range: For VLAN ID, 1 through 511 for normal interfaces, and 512 through 4094 for VLAN CCC interfaces. VLAN ID 0 is reserved for tagging the priority of frames.

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

Related Documentation
- *Configuring Dual VLAN Tags*
- stacked-vlan-tagging on page 565

vpi (Define Virtual Path)

Syntax

```
vpi vpi-identifier {
  maximum-vcs maximum-vcs;
  oam-liveness {
    up-count cells;
    down-count cells;
  }
  oam-period (disable | seconds);
  shaping {
    (cbr rate | rtvbr peak rate sustained rate burst length | vbr peak rate sustained rate burst
      length);
    queue-length number;
  }
}
```

Hierarchy Level [edit interfaces at-*fpc/pic/port* atm-options]

Release Information Statement introduced before Junos OS Release 7.4.

Description For ATM interfaces, configure the virtual path (VP).

NOTE: **Certain options apply only to specific platforms.**

Options *vpi-identifier*—ATM virtual path identifier. This is one of the VPIs that you define in the vci statement. (For a list of hierarchy levels at which you can include the **vci** statement, see vci.)

Range: 0 through 255

The remaining statements are explained separately.

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

Related Documentation

- *Configuring the Maximum Number of ATM1 VCs on a VP*
- *multipoint-destination*
- *promiscuous-mode*
- vci on page 592

CHAPTER 40

Operational Commands

clear auto-configuration interfaces

Syntax **clear auto-configuration interfaces** ***interface-name***

Release Information Command introduced in Junos OS Release 9.5.

Description Clear dynamically created VLAN interfaces.

NOTE: For the clear command to be successful, no interface bindings (for example, DHCP server bindings) can exist on the dynamic interface.

Options ***interface-name***—Name of a physical or logical interface.

Required Privilege Level view

Related Documentation

- *Dynamic VLANs Feature Guide for Subscriber Access*
- Verifying and Managing Dynamic VLAN Configuration on page 17

List of Sample Output
clear auto-configuration interfaces (All Interfaces) on page 602
clear auto-configuration interfaces (Single Dynamically Created Interface) on page 602

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

Sample Output

clear auto-configuration interfaces (All Interfaces)

```
user@host> clear auto-configuration interfaces ge-1/0/0
10 interfaces removed from device ge-1/0/0
```

clear auto-configuration interfaces (Single Dynamically Created Interface)

```
user@host> clear auto-configuration interfaces ge-1/0/0.1073741824
Interface ge-1/0/0.1073741824 deleted
```


clear auto-configuration interfaces interface-set

Syntax clear auto-configuration interfaces interface-set *interface-set-name*

Release Information Command introduced in Junos OS Release 12.2.

Description Clear a specified dynamic agent circuit identifier (ACI) interface set on the router. An ACI interface set is a logical collection of dynamic VLAN subscriber interfaces that originate at the same household or on the same access-loop port.

You can clear only those ACI interface sets that have no active subscriber interface members. If the ACI interface set that you want to clear still has valid member interfaces, you must first remove these interfaces before issuing the **clear auto-configuration interfaces interface-set *interface-set-name*** command.

Options *interface-set-name*—Name of the empty ACI interface set that you want to clear. Use the ACI interface set name generated by the router, such as aci-1003-ge-1/0/0.4001, and not the actual agent circuit identifier string found in the DHCP or PPPoE control packets. To view the names of the ACI interface sets configured on the router, you can issue the **show subscribers** command.

Required Privilege Level view

Related Documentation
- Clearing Agent Circuit Identifier Interface Sets on page 39

List of Sample Output
clear auto-configuration interfaces interface-set on page 603
clear auto-configuration interfaces interface-set (Error Message for ACI Interface Set with Active Members) on page 603

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

Sample Output

clear auto-configuration interfaces interface-set

```
user@host> clear auto-configuration interfaces interface-set aci-1003-ge-1/0/0.4001
Interface-set aci-1003-ge-1/0/0.4001 deleted
```

clear auto-configuration interfaces interface-set (Error Message for ACI Interface Set with Active Members)

```
user@host> clear auto-configuration interfaces interface-set aci-1005-ge-1/0/0.2800
error: Interface set aci-1005-ge-1/0/0.2800 has references.
```


show dhcp server binding

Syntax

show dhcp server binding
<*address*>
<*interfaces-vlan*><brief | detail | summary>
<interface *interface-name*>
<*interfaces-vlan*>
<*interfaces-wildcard*>
<logical-system *logical-system-name*>
<routing-instance *routing-instance-name*>

Release Information

Command introduced in Junos OS Release 9.0.
Options *interfaces-vlan* and *interfaces-wildcard* added in Junos OS Release 12.1.

Description

Display the address bindings in the client table on the extended Dynamic Host Configuration Protocol (DHCP) local server.

NOTE: If you delete the DHCP server configuration, DHCP server bindings might still remain. To ensure that DHCP bindings are removed, issue the **clear dhcp server binding** command before you delete the DHCP server configuration.

Options

address—(Optional) Display DHCP binding information for a specific client identified by one of the following entries:

- *ip-address*—The specified IP address.
- *mac-address*—The specified MAC address.
- *session-id*—The specified session ID.

brief | detail | summary—(Optional) Display the specified level of output about active client bindings. The default is **brief**, which produces the same output as **show dhcp server binding**.

interface *interface-name*—(Optional) Display information about active client bindings on the specified interface. You can optionally filter on VLAN ID and SVLAN ID.

interfaces-vlan—(Optional) Show the binding state information on the interface VLAN ID and S-VLAN ID.

interfaces-wildcard—(Optional) The set of interfaces on which to show the binding state information. This option supports the use of the wildcard character (*).

logical-system *logical-system-name*—(Optional) Display information about active client bindings for DHCP clients on the specified logical system.

routing-instance *routing-instance-name*—(Optional) Display information about active client bindings for DHCP clients on the specified routing instance.

Required Privilege Level view

Related Documentation

- *Clearing DHCP Bindings for Subscriber Access*
- Verifying and Managing Agent Circuit Identifier-Based Dynamic VLAN Configuration on page 38
- *clear dhcp server binding*

List of Sample Output

Output Fields Table 11 on page 605 lists the output fields for the **show dhcp server binding** command. Output fields are listed in the approximate order in which they appear.

Table 11: show dhcp server binding Output Fields

Field Name	Field Description	Level of Output
number clients, (*number* init, *number* bound, *number* selecting, *number* requesting, *number* renewing, *number* releasing)	Summary counts of the total number of DHCP clients and the number of DHCP clients in each state.	**summary**
IP address	IP address of the DHCP client.	**brief** **detail**
Session Id	Session ID of the subscriber session.	**brief** **detail**
Hardware address	Hardware address of the DHCP client.	**brief** **detail**
Expires	Number of seconds in which lease expires.	**brief** **detail**

Table 11: show dhcp server binding Output Fields *(continued)*

Field Name	Field Description	Level of Output
State	State of the address binding table on the extended DHCP local server: • **BOUND**—Client has active IP address lease. • **FORCERENEW**—Client has received forcerenew message from server. • **INIT**—Initial state. • **RELEASE**—Client is releasing IP address lease. • **RENEWING**—Client sending request to renew IP address lease. • **REQUESTING**—Client requesting a DHCP server. • **SELECTING**—Client receiving offers from DHCP servers.	**brief** **detail**
Interface	Interface on which the request was received.	**brief**
Lease Expires	Date and time at which the client's IP address lease expires.	**detail**
Lease Expires in	Number of seconds in which lease expires.	**detail**
Lease Start	Date and time at which the client's IP address lease started.	**detail**
Lease time violated	Lease time violation has occurred.	**detail**
Last Packet Received	Date and time at which the router received the last packet.	**detail**
Incoming Client Interface	Client's incoming interface.	**detail**
Client Interface Svlan Id	S-VLAN ID of the client's incoming interface.	**detail**
Client Interface Vlan Id	VLAN ID of the client's incoming interface.	**detail**
Demux Interface	Name of the IP demultiplexing (demux) interface.	**detail**
Server IP Address or **Server Identifier**	IP address of DHCP server.	**detail**
Server Interface	Interface of DHCP server.	**detail**
Client Pool Name	Name of address pool used to assign client IP address lease.	**detail**
ACI Interface Set Name	Internally generated name of the dynamic agent circuit identifier (ACI) interface set.	**detail**
ACI Interface Set Index	Index number of the dynamic ACI interface set.	**detail**
ACI Interface Set Session ID	Identifier of the dynamic ACI interface set entry in the session database.	**detail**

Sample Output

show dhcp server binding

```
user@host> show dhcp server binding
IP address        Session Id  Hardware address   Expires     State      Interface
100.20.20.15      6          00:10:94:00:00:01  86180       BOUND      ge-1/0/0.0

100.20.20.16      7          00:10:94:00:00:02  86180       BOUND      ge-1/0/0.0

100.20.20.17      8          00:10:94:00:00:03  86180       BOUND      ge-1/0/0.0

100.20.20.18      9          00:10:94:00:00:04  86180       BOUND      ge-1/0/0.0

100.20.20.19      10         00:10:94:00:00:05  86180       BOUND      ge-1/0/0.0
```

show dhcp server binding detail

```
user@host> show dhcp server binding detail
Client IP Address:  100.20.20.15
     Hardware Address:             00:10:94:00:00:01
     State:                       BOUND(LOCAL_SERVER_STATE_BOUND_ON_INTF_DELETE)

     Lease Expires:                2009-07-21 10:10:25 PDT
     Lease Expires in:             86151 seconds
     Lease Start:                  2009-07-20 10:10:25 PDT
     Incoming Client Interface:    ge-1/0/0.0
     Server Ip Address:            100.20.20.9
     Server Interface:             none
     Session Id:                   6
     Client Pool Name:             6
     Client IP Address:            100.20.20.16
     Hardware Address:             00:10:94:00:00:02
     State:                       BOUND(LOCAL_SERVER_STATE_BOUND_ON_INTF_DELETE)

     Lease Expires:                2009-07-21 10:10:25 PDT
     Lease Expires in:             86151 seconds
     Lease Start:                  2009-07-20 10:10:25 PDT
     Lease time violated:          yes
     Incoming Client Interface:    ge-1/0/0.0
     Server Ip Address:            100.20.20.9
     Server Interface:             none
     Session Id:                   7
     Client Pool Name:             7
```

show dhcp server binding detail (ACI Interface Set Configured)

```
user@host> show dhcp server binding detail
Client IP Address:  100.20.22.14
     Hardware Address:             00:00:64:34:01:02
     State:                        BOUND(LOCAL_SERVER_STATE_BOUND)
     Lease Expires:                2012-03-13 09:53:32 PDT
     Lease Expires in:             82660 seconds
     Lease Start:                  2012-03-12 10:23:32 PDT
     Last Packet Received:         2012-03-12 10:23:32 PDT
     Incoming Client Interface:    demux0.1073741827
     Client Interface Svlan Id:    1802
     Client Interface Vlan Id:     302
     Demux Interface:              demux0.1073741832
     Server Identifier:            100.20.200.202
     Session Id:                   11
```



```
Client Pool Name:             poolA
Client Profile Name:          DEMUXprofile
ACI Interface Set Name:       aci-1002-demux0.1073741827
ACI Interface Set Index:      2
ACI Interface Set Session ID: 6
```

show dhcp server binding interface <vlan-id>

```
user@host> show dhcp server binding interface ge-1/1/0:100
IP address        Session Id  Hardware address   Expires     State      Interface
200.20.20.15       6           00:10:94:00:00:01  86124       BOUND
ge-1/1/0:100
```

show dhcp server binding interface <svlan-id>

```
user@host> show dhcp server binding interface ge-1/1/0:10-100
IP address        Session Id  Hardware address   Expires     State      Interface
200.20.20.16       7           00:10:94:00:00:02  86124       BOUND
ge-1/1/0:10-100
```

show dhcp server binding <ip-address>

```
user@host> show dhcp server binding 100.20.20.19
IP address        Session Id  Hardware address   Expires     State      Interface
100.20.20.19      10          00:10:94:00:00:05  86081       BOUND      ge-1/0/0.0
```

show dhcp server binding <session-id>

```
user@host> show dhcp server binding 6
IP address        Session Id  Hardware address   Expires     State      Interface
200.20.20.15      6           00:10:94:00:00:01  86124       BOUND      ge-1/0/0.0
```

show dhcp server binding summary

```
user@host> show dhcp server binding summary
3 clients, (2 init, 1 bound, 0 selecting, 0 requesting, 0 renewing, 0 releasing)
```

show dhcp server binding <interfaces-vlan>

```
user@host> show dhcp server binding ge-1/0/0:100-200
IP address        Session Id  Hardware address   Expires     State      Interface
192.168.0.17       42          00:10:94:00:00:02  86346       BOUND
ge-1/0/0.1073741827
192.168.0.16       41          00:10:94:00:00:01  86346       BOUND
ge-1/0/0.1073741827
```

show dhcp server binding <interfaces-wildcard>

```
user@host> show dhcp server binding ge-1/3/*
IP address        Session Id  Hardware address   Expires     State      Interface
192.168.0.9        24          00:10:94:00:00:04  86361       BOUND
ge-1/3/0.110
192.168.0.8        23          00:10:94:00:00:03  86361       BOUND
ge-1/3/0.110
192.168.0.7        22          00:10:94:00:00:02  86361       BOUND
ge-1/3/0.110
```


clear pppoe lockout

Syntax

clear pppoe lockout
<mac-address *mac-address*>
<aci *circuit-id*>
<underlying-interfaces *underlying-interface-name*>

Release Information

Command introduced in Junos OS Release 11.4.

Description

Clear the lockout condition for the PPPoE client associated with the specified MAC source address.

Options

none—Clear the lockout condition for the PPPoE clients associated with all MAC source addresses on all PPPoE underlying interfaces.

aci—(Optional) Clear the lockout condition for the PPPoE client associated with the specified agent circuit identifier (ACI).

mac-address ***mac-address***—(Optional) Clear the lockout condition for the PPPoE client associated with the specified MAC source address.

underlying-interfaces ***underlying-interface-name***—(Optional) Clear the lockout condition for all PPPoE clients associated with the specified PPPoE underlying interface.

Required Privilege Level

clear

Related Documentation

- Clearing Lockout of PPPoE Subscriber Sessions on page 199
- Configuring Lockout of PPPoE Subscriber Sessions on page 198

List of Sample Output

Sample Output

clear pppoe lockout aci (Clear agent circuit identifier (ACI) on All Underlying Interfaces)

`user@host>` **clear pppoe lockout aci**

clear pppoe lockout aci quoted-aci-regexp(Clear agent circuit identifier (ACI) on Specified Underlying Interface)

`user@host>` **clear pppoe lockout underlying-interfaces demux0.214 aci "Relay-identifier atm 3/0:100\.*"**

clear pppoe lockout (Clear All MAC Source Addresses on All Underlying Interfaces)

`user@host>` **clear pppoe lockout**

clear pppoe lockout mac-address (Clear Specified MAC Source Address)

`user@host>` **clear pppoe lockout mac-address 00:1d:72:bc:53:30**

clear pppoe lockout underlying-interfaces (Clear All MAC Source Addresses on Specified Underlying Interface)

`user@host>` **clear pppoe lockout underlying-interfaces ge-1/0/0.101**

clear pppoe lockout mac-address underlying-interfaces (Clear Specified MAC Source Address on Specified Underlying Interface)

`user@host>` **clear pppoe lockout mac-address 00:1d:72:bc:53:30 underlying-interfaces ge-1/0/0.101**

clear pppoe statistics

Syntax

clear pppoe statistics
<interface *interface-name*>
<*underlying-interface-name*>

Release Information

Command introduced before Junos OS Release 7.4.
underlying-interface-name option introduced in Junos OS Release 9.5.

Description

(J Series routers, M120 routers, M320 routers, and MX Series routers only) Reset PPPoE session statistics information.

Options

none—Reset PPPoE statistics for all interfaces.

interface *interface-name*—(J Series routers) (Optional) Reset PPPoE statistics for the specified interface.

underlying-interface-name—(M120 routers, M320 routers, and MX Series routers) (Optional) Reset PPPoE statistics for the specified underlying PPPoE interface.

Required Privilege Level

clear

Related Documentation

- show pppoe statistics on page 741

List of Sample Output

clear pppoe statistics on page 611
clear pppoe statistics interface (PPPoE Interfaces on J Series Routers) on page 611
clear pppoe statistics (PPPoE Underlying Interfaces on M Series and MX Series Routers) on page 611

Output Fields

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

Sample Output

clear pppoe statistics

```
user@host> clear pppoe statistics
```

clear pppoe statistics interface (PPPoE Interfaces on J Series Routers)

```
user@host> clear pppoe statistics interface pp0.1073741827
```

clear pppoe statistics (PPPoE Underlying Interfaces on M Series and MX Series Routers)

```
user@host> clear pppoe statistics ge-4/0/3.2
```


show interfaces (10-Gigabit Ethernet)

Syntax

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

Release Information

Command introduced in Junos OS Release 8.0.

Description

(M320, M120, MX Series, and T Series routers and EX Series switches only) Display status information about the specified 10-Gigabit Ethernet interface.

Options

xe-*fpc/pic/port*—Display standard information about the specified 10-Gigabit Ethernet interface.

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

descriptions—(Optional) Display interface description strings.

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

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

statistics—(Optional) Display static interface statistics.

Required Privilege Level

view

List of Sample Output

show interfaces extensive (10-Gigabit Ethernet, LAN PHY Mode, IQ2) on page 627
show interfaces extensive (10-Gigabit Ethernet, WAN PHY Mode) on page 630
show interfaces extensive (10-Gigabit Ethernet, DWDM OTN PIC) on page 632
show interfaces extensive (10-Gigabit Ethernet, LAN PHY Mode, Unidirectional Mode) on page 634
show interfaces extensive (10-Gigabit Ethernet, LAN PHY Mode, Unidirectional Mode, Transmit-Only) on page 634
show interfaces extensive (10-Gigabit Ethernet, LAN PHY Mode, Unidirectional Mode, Receive-Only) on page 635

Output Fields

See Table 12 on page 613 for the output fields for the **show interfaces** (10–Gigabit Ethernet) command.

Table 12: show interfaces Gigabit Ethernet Output Fields

Field Name	Field Description	Level of Output
Physical Interface		
Physical interface	Name of the physical interface.	All levels
Enabled	State of the interface. Possible values are described in the “Enabled Field” section under *Common Output Fields Description*.	All levels
Interface index	Index number of the physical interface, which reflects its initialization sequence.	**detail extensive** none
SNMP ifIndex	SNMP index number for the physical interface.	**detail extensive** none
Generation	Unique number for use by Juniper Networks technical support only.	**detail extensive**
Link-level type	Encapsulation being used on the physical interface.	All levels
MTU	Maximum transmission unit size on the physical interface.	All levels
Speed	Speed at which the interface is running.	All levels
Loopback	Loopback status: **Enabled** or **Disabled**. If loopback is enabled, type of loopback: **Local** or **Remote**.	All levels
Source filtering	Source filtering status: **Enabled** or **Disabled**.	All levels
LAN-PHY mode	10-Gigabit Ethernet interface operating in Local Area Network Physical Layer Device (LAN PHY) mode. LAN PHY allows 10-Gigabit Ethernet wide area links to use existing Ethernet applications.	All levels
WAN-PHY mode	10-Gigabit Ethernet interface operating in Wide Area Network Physical Layer Device (WAN PHY) mode. WAN PHY allows 10-Gigabit Ethernet wide area links to use fiber-optic cables and other devices intended for SONET/SDH.	All levels
Unidirectional	Unidirectional link mode status for 10-Gigabit Ethernet interface: **Enabled** or **Disabled** for parent interface; **Rx-only** or **Tx-only** for child interfaces.	All levels
Flow control	Flow control status: **Enabled** or **Disabled**.	All levels
Auto-negotiation	(Gigabit Ethernet interfaces) Autonegotiation status: **Enabled** or **Disabled**.	All levels
Remote-fault	(Gigabit Ethernet interfaces) Remote fault status: • **Online**—Autonegotiation is manually configured as online. • **Offline**—Autonegotiation is manually configured as offline.	All levels
Device flags	Information about the physical device. Possible values are described in the “Device Flags” section under *Common Output Fields Description*.	All levels
Interface flags	Information about the interface. Possible values are described in the “Interface Flags” section under *Common Output Fields Description*.	All levels

Table 12: show interfaces Gigabit Ethernet Output Fields *(continued)*

Field Name	Field Description	Level of Output	
Link flags	Information about the link. Possible values are described in the "Links Flags" section under *Common Output Fields Description*.	All levels	
Wavelength	(10-Gigabit Ethernet dense wavelength-division multiplexing [DWDM] interfaces) Displays the configured wavelength, in nanometers (nm).	All levels	
Frequency	(10-Gigabit Ethernet DWDM interfaces only) Displays the frequency associated with the configured wavelength, in terahertz (THz).	All levels	
CoS queues	Number of CoS queues configured.	**detail extensive** none	
Schedulers	(Gigabit Ethernet intelligent queuing 2 (IQ2) interfaces only) Number of CoS schedulers configured.	**extensive**	
Hold-times	Current interface hold-time up and hold-time down, in milliseconds.	**detail extensive**	
Current address	Configured MAC address.	**detail extensive** none	
Hardware address	Hardware MAC address.	**detail extensive** none	
Last flapped	Date, time, and how long ago the interface went from down to up. The format is **Last flapped:** ***year-month-day hour*****:*****minute*****:*****second*****:*****timezone*** **(*****hour*****:*****minute*****:*****second*** **ago)**. For example, **Last flapped: 2002-04-26 10:52:40 PDT (04:33:20 ago).**	**detail extensive** none	
Input Rate	Input rate in bits per second (bps) and packets per second (pps). The value in this field also includes the Layer 2 overhead bytes for ingress traffic on Ethernet interfaces if you enable accounting of Layer 2 overhead at the PIC level or the logical interface level.	None specified	
Output Rate	Output rate in bps and pps. The value in this field also includes the Layer 2 overhead bytes for egress traffic on Ethernet interfaces if you enable accounting of Layer 2 overhead at the PIC level or the logical interface level.	None specified	
Statistics last cleared	Time when the statistics for the interface were last set to zero.	**detail extensive**	
Egress account overhead	Layer 2 overhead in bytes that is accounted in the interface statistics for egress traffic.	**detail extensive**	
Ingress account overhead	Layer 2 overhead in bytes that is accounted in the interface statistics for ingress traffic.	**detail extensive**	**detail extensive**

Table 12: show interfaces Gigabit Ethernet Output Fields *(continued)*

Field Name	Field Description	Level of Output
Traffic statistics	Number and rate of bytes and packets received and transmitted on the physical interface. • **Input bytes**—Number of bytes received on the interface. The value in this field also includes the Layer 2 overhead bytes for ingress traffic on Ethernet interfaces if you enable accounting of Layer 2 overhead at the PIC level or the logical interface level. • **Output bytes**—Number of bytes transmitted on the interface. The value in this field also includes the Layer 2 overhead bytes for egress traffic on Ethernet interfaces if you enable accounting of Layer 2 overhead at the PIC level or the logical interface level. • **Input packets**—Number of packets received on the interface. • **Output packets**—Number of packets transmitted on the interface. Gigabit Ethernet and 10-Gigabit Ethernet IQ PICs count the overhead and CRC bytes. For Gigabit Ethernet IQ PICs, the input byte counts vary by interface type. For more information, see Table 12 on page 613.	**detail extensive**
Input errors	Input errors on the interface. The following paragraphs explain the counters whose meaning might not be obvious: • **Errors**—Sum of the incoming frame aborts and FCS errors. • **Drops**—Number of packets dropped by the input queue of the I/O Manager ASIC. If the interface is saturated, this number increments once for every packet that is dropped by the ASIC's RED mechanism. • **Framing errors**—Number of packets received with an invalid frame checksum (FCS). • **Runts**—Number of frames received that are smaller than the runt threshold. • **Policed discards**—Number of frames that the incoming packet match code discarded because they were not recognized or not of interest. Usually, this field reports protocols that the Junos OS does not handle. • **L3 incompletes**—Number of incoming packets discarded because they failed Layer 3 (usually IPv4) sanity checks of the header. For example, a frame with less than 20 bytes of available IP header is discarded. L3 incomplete errors can be ignored by configuring the **ignore-l3-incompletes** statement. • **L2 channel errors**—Number of times the software did not find a valid logical interface for an incoming frame. • **L2 mismatch timeouts**—Number of malformed or short packets that caused the incoming packet handler to discard the frame as unreadable. • **FIFO errors**—Number of FIFO errors in the receive direction that are reported by the ASIC on the PIC. If this value is ever nonzero, the PIC is probably malfunctioning. • **Resource errors**—Sum of transmit drops.	**extensive**

Table 12: show interfaces Gigabit Ethernet Output Fields *(continued)*

Field Name	Field Description	Level of Output
Output errors	Output errors on the interface. The following paragraphs explain the counters whose meaning might not be obvious: • **Carrier transitions**—Number of times the interface has gone from **down** to **up**. This number does not normally increment quickly, increasing only when the cable is unplugged, the far-end system is powered down and then up, or another problem occurs. If the number of carrier transitions increments quickly (perhaps once every 10 seconds), the cable, the far-end system, or the PIC or PIM is malfunctioning. • **Errors**—Sum of the outgoing frame aborts and FCS errors. • **Drops**—Number of packets dropped by the output queue of the I/O Manager ASIC. If the interface is saturated, this number increments once for every packet that is dropped by the ASIC's RED mechanism. • **Collisions**—Number of Ethernet collisions. The Gigabit Ethernet PIC supports only full-duplex operation, so for Gigabit Ethernet PICs, this number should always remain 0. If it is nonzero, there is a software bug. • **Aged packets**—Number of packets that remained in shared packet SDRAM so long that the system automatically purged them. The value in this field should never increment. If it does, it is most likely a software bug or possibly malfunctioning hardware. • **FIFO errors**—Number of FIFO errors in the send direction as reported by the ASIC on the PIC. If this value is ever nonzero, the PIC is probably malfunctioning. • **HS link CRC errors**—Number of errors on the high-speed links between the ASICs responsible for handling the router interfaces. • **MTU errors**—Number of packets whose size exceeded the MTU of the interface. • **Resource errors**—Sum of transmit drops.	**extensive**
Egress queues	Total number of egress queues supported on the specified interface.	**detail extensive**
Queue counters (Egress)	CoS queue number and its associated user-configured forwarding class name. • **Queued packets**—Number of queued packets. • **Transmitted packets**—Number of transmitted packets. • **Dropped packets**—Number of packets dropped by the ASIC's RED mechanism.	**detail extensive**
Ingress queues	Total number of ingress queues supported on the specified interface. Displayed on IQ2 interfaces.	**extensive**
Queue counters (Ingress)	CoS queue number and its associated user-configured forwarding class name. Displayed on IQ2 interfaces. • **Queued packets**—Number of queued packets. • **Transmitted packets**—Number of transmitted packets. • **Dropped packets**—Number of packets dropped by the ASIC's RED mechanism.	**extensive**

Table 12: show interfaces Gigabit Ethernet Output Fields *(continued)*

Field Name	Field Description	Level of Output
Active alarms and Active defects	Ethernet-specific defects that can prevent the interface from passing packets. When a defect persists for a certain amount of time, it is promoted to an alarm. Based on the routing device configuration, an alarm can ring the red or yellow alarm bell on the routing device, or turn on the red or yellow alarm LED on the craft interface. These fields can contain the value **None** or **Link**. • **None**—There are no active defects or alarms. • **Link**—Interface has lost its link state, which usually means that the cable is unplugged, the far-end system has been turned off, or the PIC is malfunctioning.	**detail extensive** none
OTN alarms	Active OTN alarms identified on the interface.	**detail extensive**
OTN defects	OTN defects received on the interface.	**detail extensive**
OTN FEC Mode	The FECmode configured on the interface. • **efec**—Enhanced forward error correction (EFEC) is configured to defect and correct bit errors. • **gfec**—G.709 Forward error correction (GFEC) mode is configured to detect and correct bit errors. • **none**—FEC mode is not configured.	**detail extensive**
OTN Rate	OTN mode. • **fixed-stuff-bytes**—Fixed stuff bytes 11.0957 Gbps. • **no-fixed-stuff-bytes**—No fixed stuff bytes 11.0491 Gbps. • **pass-through**—Enable OTN passthrough mode. • **no-pass-through**—Do not enable OTN passthrough mode.	**detail extensive**
OTN Line Loopback	Status of the line loopback, if configured for the DWDM OTN PIC. Its value can be: **enabled** or **disabled**.	**detail extensive**
OTN FEC statistics	The forward error correction (FEC) counters for the DWDM OTN PIC. • **Corrected Errors**—The count of corrected errors in the last second. • **Corrected Error Ratio**—The corrected error ratio in the last 25 seconds. For example, 1e-7 is 1 error per 10 million bits.	**detail extensive**
OTN FEC alarms	OTN FEC excessive or degraded error alarms triggered on the interface. • **FEC Degrade**—OTU FEC Degrade defect. • **FEC Excessive**—OTU FEC Excessive Error defect.	**detail extensive**
OTN OC	OTN OC defects triggered on the interface. • **LOS**—OC Loss of Signal defect. • **LOF**—OC Loss of Frame defect. • **LOM**—OC Loss of Multiframe defect. • **Wavelength Lock**—OC Wavelength Lock defect.	**detail extensive**

Table 12: show interfaces Gigabit Ethernet Output Fields *(continued)*

Field Name	Field Description	Level of Output
OTN OTU	OTN OTU defects detected on the interface • **AIS**—OTN AIS alarm. • **BDI**—OTN OTU BDI alarm. • **IAE**—OTN OTU IAE alarm. • **TTIM**—OTN OTU TTIM alarm. • **SF**—OTN ODU bit error rate fault alarm. • **SD**—OTN ODU bit error rate defect alarm. • **TCA-ES**—OTN ODU ES threshold alarm. • **TCA-SES**—OTN ODU SES threshold alarm. • **TCA-UAS**—OTN ODU UAS threshold alarm. • **TCA-BBE**—OTN ODU BBE threshold alarm. • **BIP**—OTN ODU BIP threshold alarm. • **BBE**—OTN OTU BBE threshold alarm. • **ES**—OTN OTU ES threshold alarm. • **SES**—OTN OTU SES threshold alarm. • **UAS**—OTN OTU UAS threshold alarm.	**detail extensive**
Received DAPI	Destination Access Port Interface (DAPI) from which the packets were received.	**detail extensive**
Received SAPI	Source Access Port Interface (SAPI) from which the packets were received.	**detail extensive**
Transmitted DAPI	Destination Access Port Interface (DAPI) to which the packets were transmitted.	**detail extensive**
Transmitted SAPI	Source Access Port Interface (SAPI) to which the packets were transmitted.	**detail extensive**
PCS statistics	(10-Gigabit Ethernet interfaces) Displays Physical Coding Sublayer (PCS) fault conditions from the WAN PHY or the LAN PHY device. • **Bit errors**—The number of seconds during which at least one bit error rate (BER) occurred while the PCS receiver is operating in normal mode. • **Errored blocks**—The number of seconds when at least one errored block occurred while the PCS receiver is operating in normal mode.	**detail extensive**

Table 12: show interfaces Gigabit Ethernet Output Fields *(continued)*

Field Name	Field Description	Level of Output
MAC statistics	**Receive** and **Transmit** statistics reported by the PIC's MAC subsystem, including the following: • **Total octets** and **total packets**—Total number of octets and packets. For Gigabit Ethernet IQ PICs, the received octets count varies by interface type. For more information, see Table 13 on page 627 • **Unicast packets, Broadcast packets,** and **Multicast packets**—Number of unicast, broadcast, and multicast packets. • **CRC/Align errors**—Total number of packets received that had a length (excluding framing bits, but including FCS octets) of between 64 and 1518 octets, inclusive, and had either a bad FCS with an integral number of octets (FCS Error) or a bad FCS with a nonintegral number of octets (Alignment Error). • **FIFO error**—Number of FIFO errors that are reported by the ASIC on the PIC. If this value is ever nonzero, the PIC or a cable is probably malfunctioning. • **MAC control frames**—Number of MAC control frames. • **MAC pause frames**—Number of MAC control frames with **pause** operational code. • **Oversized frames**—Number of frames that exceed 1518 octets. • **Jabber frames**—Number of frames that were longer than 1518 octets (excluding framing bits, but including FCS octets), and had either an FCS error or an alignment error. This definition of jabber is different from the definition in IEEE-802.3 section 8.2.1.5 (10BASE5) and section 10.3.1.4 (10BASE2). These documents define jabber as the condition in which any packet exceeds 20 ms. The allowed range to detect jabber is from 20 ms to 150 ms. • **Fragment frames**—Total number of packets that were less than 64 octets in length (excluding framing bits, but including FCS octets), and had either an FCS error or an alignment error. Fragment frames normally increment because both runts (which are normal occurrences caused by collisions) and noise hits are counted. • **VLAN tagged frames**—Number of frames that are VLAN tagged. The system uses the TPID of 0x8100 in the frame to determine whether a frame is tagged or not. • **Code violations**—Number of times an event caused the PHY to indicate "Data reception error" or "invalid data symbol error."	**extensive**
OTN Received Overhead Bytes	APS/PCC0: 0x02, APS/PCC1: 0x11, APS/PCC2: 0x47, APS/PCC3: 0x58 Payload Type: 0x08	**extensive**
OTN Transmitted Overhead Bytes	APS/PCC0: 0x00, APS/PCC1: 0x00, APS/PCC2: 0x00, APS/PCC3: 0x00 Payload Type: 0x08	**extensive**

Table 12: show interfaces Gigabit Ethernet Output Fields *(continued)*

Field Name	Field Description	Level of Output
Filter statistics	**Receive** and **Transmit** statistics reported by the PIC's MAC address filter subsystem. The filtering is done by the content-addressable memory (CAM) on the PIC. The filter examines a packet's source and destination MAC addresses to determine whether the packet should enter the system or be rejected. • **Input packet count**—Number of packets received from the MAC hardware that the filter processed. • **Input packet rejects**—Number of packets that the filter rejected because of either the source MAC address or the destination MAC address. • **Input DA rejects**—Number of packets that the filter rejected because the destination MAC address of the packet is not on the accept list. It is normal for this value to increment. When it increments very quickly and no traffic is entering the routing device from the far-end system, either there is a bad ARP entry on the far-end system, or multicast routing is not on and the far-end system is sending many multicast packets to the local routing device (which the routing device is rejecting). • **Input SA rejects**—Number of packets that the filter rejected because the source MAC address of the packet is not on the accept list. The value in this field should increment only if source MAC address filtering has been enabled. If filtering is enabled, if the value increments quickly, and if the system is not receiving traffic that it should from the far-end system, it means that the user-configured source MAC addresses for this interface are incorrect. • **Output packet count**—Number of packets that the filter has given to the MAC hardware. • **Output packet pad count**—Number of packets the filter padded to the minimum Ethernet size (60 bytes) before giving the packet to the MAC hardware. Usually, padding is done only on small ARP packets, but some very small IP packets can also require padding. If this value increments rapidly, either the system is trying to find an ARP entry for a far-end system that does not exist or it is misconfigured. • **Output packet error count**—Number of packets with an indicated error that the filter was given to transmit. These packets are usually aged packets or are the result of a bandwidth problem on the FPC hardware. On a normal system, the value of this field should not increment. • **CAM destination filters, CAM source filters**—Number of entries in the CAM dedicated to destination and source MAC address filters. There can only be up to 64 source entries. If source filtering is disabled, which is the default, the values for these fields should be 0.	**extensive**
PMA PHY	(10-Gigabit Ethernet interfaces, WAN PHY mode) SONET error information: • **Seconds**—Number of seconds the defect has been active. • **Count**—Number of times that the defect has gone from inactive to active. • **State**—State of the error. Any state other than **OK** indicates a problem. Subfields are: • **PHY Lock**—Phase-locked loop • **PHY Light**—Loss of optical signal	**extensive**

Table 12: show interfaces Gigabit Ethernet Output Fields *(continued)*

Field Name	Field Description	Level of Output
WIS section	(10-Gigabit Ethernet interfaces, WAN PHY mode) SONET error information: • **Seconds**—Number of seconds the defect has been active. • **Count**—Number of times that the defect has gone from inactive to active. • **State**—State of the error. Any state other than **OK** indicates a problem. Subfields are: • **BIP-B1**—Bit interleaved parity for SONET section overhead • **SEF**—Severely errored framing • **LOL**—Loss of light • **LOF**—Loss of frame • **ES-S**—Errored seconds (section) • **SES-S**—Severely errored seconds (section) • **SEFS-S**—Severely errored framing seconds (section)	**extensive**
WIS line	(10-Gigabit Ethernet interfaces, WAN PHY mode) Active alarms and defects, plus counts of specific SONET errors with detailed information. • **Seconds**—Number of seconds the defect has been active. • **Count**—Number of times that the defect has gone from inactive to active. • **State**—State of the error. State other than **OK** indicates a problem. Subfields are: • **BIP-B2**—Bit interleaved parity for SONET line overhead • **REI-L**—Remote error indication (near-end line) • **RDI-L**—Remote defect indication (near-end line) • **AIS-L**—Alarm indication signal (near-end line) • **BERR-SF**—Bit error rate fault (signal failure) • **BERR-SD**—Bit error rate defect (signal degradation) • **ES-L**—Errored seconds (near-end line) • **SES-L**—Severely errored seconds (near-end line) • **UAS-L**—Unavailable seconds (near-end line) • **ES-LFE**—Errored seconds (far-end line) • **SES-LFE**—Severely errored seconds (far-end line) • **UAS-LFE**—Unavailable seconds (far-end line)	**extensive**

Table 12: show interfaces Gigabit Ethernet Output Fields *(continued)*

Field Name	Field Description	Level of Output
WIS path	(10-Gigabit Ethernet interfaces, WAN PHY mode) Active alarms and defects, plus counts of specific SONET errors with detailed information. • **Seconds**—Number of seconds the defect has been active. • **Count**—Number of times that the defect has gone from inactive to active. • **State**—State of the error. Any state other than **OK** indicates a problem. Subfields are: • **BIP-B3**—Bit interleaved parity for SONET section overhead • **REI-P**—Remote error indication • **LOP-P**—Loss of pointer (path) • **AIS-P**—Path alarm indication signal • **RDI-P**—Path remote defect indication • **UNEQ-P**—Path unequipped • **PLM-P**—Path payload label mismatch • **ES-P**—Errored seconds (near-end STS path) • **SES-P**—Severely errored seconds (near-end STS path) • **UAS-P**—Unavailable seconds (near-end STS path) • **SES-PFE**—Severely errored seconds (far-end STS path) • **UAS-PFE**—Unavailable seconds (far-end STS path)	**extensive**

Table 12: show interfaces Gigabit Ethernet Output Fields *(continued)*

Field Name	Field Description	Level of Output

Autonegotiation information	Information about link autonegotiation.
	• **Negotiation status:**
	  ◦ **Incomplete**—Ethernet interface has the speed or link mode configured.
	  ◦ **No autonegotiation**—Remote Ethernet interface has the speed or link mode configured, or does not perform autonegotiation.
	  ◦ **Complete**—Ethernet interface is connected to a device that performs autonegotiation and the autonegotiation process is successful.
	• **Link partner status**—**OK** when Ethernet interface is connected to a device that performs autonegotiation and the autonegotiation process is successful.
	• **Link partner:**
	  ◦ **Link mode**—Depending on the capability of the attached Ethernet device, either **Full-duplex** or **Half-duplex**.
	  ◦ **Flow control**—Types of flow control supported by the remote Ethernet device. For Fast Ethernet interfaces, the type is **None**. For Gigabit Ethernet interfaces, types are **Symmetric** (link partner supports **PAUSE** on receive and transmit), **Asymmetric** (link partner supports **PAUSE** on transmit), and **Symmetric/Asymmetric** (link partner supports both **PAUSE** on receive and transmit or only **PAUSE** receive).
	  ◦ **Remote fault**—Remote fault information from the link partner—**Failure** indicates a receive link error. **OK** indicates that the link partner is receiving. **Negotiation error** indicates a negotiation error. **Offline** indicates that the link partner is going offline.
	• **Local resolution**—Information from the link partner:
	  ◦ **Flow control**—Types of flow control supported by the remote Ethernet device. For Gigabit Ethernet interfaces, types are **Symmetric** (link partner supports **PAUSE** on receive and transmit), **Asymmetric** (link partner supports **PAUSE** on transmit), and **Symmetric/Asymmetric** (link partner supports both **PAUSE** on receive and transmit or only **PAUSE** receive).
	  ◦ **Remote fault**—Remote fault information. **Link OK** (no error detected on receive), **Offline** (local interface is offline), and **Link Failure** (link error detected on receive).
		extensive

Received path trace, Transmitted path trace	(10-Gigabit Ethernet interfaces, WAN PHY mode) SONET/SDH interfaces allow path trace bytes to be sent inband across the SONET/SDH link. Juniper Networks and other router manufacturers use these bytes to help diagnose misconfigurations and network errors by setting the transmitted path trace message so that it contains the system hostname and name of the physical interface. The received path trace value is the message received from the routing device at the other end of the fiber. The transmitted path trace value is the message that this routing device transmits.	**extensive**
Packet Forwarding Engine configuration	Information about the configuration of the Packet Forwarding Engine: • **Destination slot**—FPC slot number.	**extensive**

Table 12: show interfaces Gigabit Ethernet Output Fields *(continued)*

Field Name	Field Description	Level of Output
CoS information	Information about the CoS queue for the physical interface. • **CoS transmit queue**—Queue number and its associated user-configured forwarding class name. • **Bandwidth %**—Percentage of bandwidth allocated to the queue. • **Bandwidth bps**—Bandwidth allocated to the queue (in bps). • **Buffer %**—Percentage of buffer space allocated to the queue. • **Buffer usec**—Amount of buffer space allocated to the queue, in microseconds. This value is nonzero only if the buffer size is configured in terms of time. • **Priority**—Queue priority: **low** or **high**. • **Limit**—Displayed if rate limiting is configured for the queue. Possible values are **none** and **exact**. If **exact** is configured, the queue transmits only up to the configured bandwidth, even if excess bandwidth is available. If **none** is configured, the queue transmits beyond the configured bandwidth if bandwidth is available.	**extensive**
Logical Interface		
Logical interface	Name of the logical interface.	All levels
Index	Index number of the logical interface, which reflects its initialization sequence.	**detail extensive** none
SNMP ifIndex	SNMP interface index number for the logical interface.	**detail extensive** none
Generation	Unique number for use by Juniper Networks technical support only.	**detail extensive**
Flags	Information about the logical interface. Possible values are described in the "Logical Interface Flags" section under *Common Output Fields Description*.	All levels

Table 12: show interfaces Gigabit Ethernet Output Fields *(continued)*

Field Name	Field Description	Level of Output
VLAN-Tag	Rewrite profile applied to incoming or outgoing frames on the outer (**Out**) VLAN tag or for both the outer and inner (**In**) VLAN tags. • **push**—An outer VLAN tag is pushed in front of the existing VLAN tag. • **pop**—The outer VLAN tag of the incoming frame is removed. • **swap**—The outer VLAN tag of the incoming frame is overwritten with the user specified VLAN tag information. • **push**—An outer VLAN tag is pushed in front of the existing VLAN tag. • **push-push**—Two VLAN tags are pushed in from the incoming frame. • **swap-push**—The outer VLAN tag of the incoming frame is replaced by a user-specified VLAN tag value. A user-specified outer VLAN tag is pushed in front. The outer tag becomes an inner tag in the final frame. • **swap-swap**—Both the inner and the outer VLAN tags of the incoming frame are replaced by the user specified VLAN tag value. • **pop-swap**—The outer VLAN tag of the incoming frame is removed, and the inner VLAN tag of the incoming frame is replaced by the user-specified VLAN tag value. The inner tag becomes the outer tag in the final frame. • **pop-pop**—Both the outer and inner VLAN tags of the incoming frame are removed.	**brief detail extensive** none
Demux:	IP demultiplexing (demux) value that appears if this interface is used as the demux underlying interface. The output is one of the following: • Source Family Inet • Destination Family Inet	**detail extensive** none
Encapsulation	Encapsulation on the logical interface.	All levels
Protocol	Protocol family. Possible values are described in the “Protocol Field” section under *Common Output Fields Description*.	**detail extensive** none
MTU	Maximum transmission unit size on the logical interface.	**detail extensive** none
Maximum labels	Maximum number of MPLS labels configured for the MPLS protocol family on the logical interface.	**detail extensive** none
Traffic statistics	Number and rate of bytes and packets received and transmitted on the specified interface set. • **Input bytes, Output bytes**—Number of bytes received and transmitted on the interface set. The value in this field also includes the Layer 2 overhead bytes for ingress or egress traffic on Ethernet interfaces if you enable accounting of Layer 2 overhead at the PIC level or the logical interface level. • **Input packets, Output packets**—Number of packets received and transmitted on the interface set.	**detail extensive**
IPv6 transit statistics	Number of IPv6 transit bytes and packets received and transmitted on the logical interface if IPv6 statistics tracking is enabled.	**extensive**
Local statistics	Number and rate of bytes and packets destined to the routing device.	**extensive**

Table 12: show interfaces Gigabit Ethernet Output Fields *(continued)*

Field Name	Field Description	Level of Output
Transit statistics	Number and rate of bytes and packets transiting the switch. NOTE: For Gigabit Ethernet intelligent queuing 2 (IQ2) interfaces, the logical interface egress statistics might not accurately reflect the traffic on the wire when output shaping is applied. Traffic management output shaping might drop packets after they are tallied by the **Output bytes** and **Output packets** interface counters. However, correct values display for both of these egress statistics when per-unit scheduling is enabled for the Gigabit Ethernet IQ2 physical interface, or when a single logical interface is actively using a shared scheduler.	**extensive**
Generation	Unique number for use by Juniper Networks technical support only.	**detail extensive**
Route Table	Route table in which the logical interface address is located. For example, **0** refers to the routing table inet.0.	**detail extensive** none
Flags	Information about protocol family flags. Possible values are described in the “Family Flags” section under *Common Output Fields Description*.	**detail extensive**
Donor interface	(Unnumbered Ethernet) Interface from which an unnumbered Ethernet interface borrows an IPv4 address.	**detail extensive** none
Preferred source address	(Unnumbered Ethernet) Secondary IPv4 address of the donor loopback interface that acts as the preferred source address for the unnumbered Ethernet interface.	**detail extensive** none
Input Filters	Names of any input filters applied to this interface. If you specify a precedence value for any filter in a dynamic profile, filter precedence values appear in parenthesis next to all interfaces.	**detail extensive**
Output Filters	Names of any output filters applied to this interface. If you specify a precedence value for any filter in a dynamic profile, filter precedence values appear in parenthesis next to all interfaces.	**detail extensive**
Mac-Validate Failures	Number of MAC address validation failures for packets and bytes. This field is displayed when MAC address validation is enabled for the logical interface.	**detail extensive** none
Addresses, Flags	Information about the address flags. Possible values are described in the “Addresses Flags” section under *Common Output Fields Description*.	**detail extensive** none
protocol-family	Protocol family configured on the logical interface. If the protocol is **inet**, the IP address of the interface is also displayed.	**brief**
Flags	Information about address flag (possible values are described in the “Addresses Flags” section under *Common Output Fields Description*.	**detail extensive** none
Destination	IP address of the remote side of the connection.	**detail extensive** none
Local	IP address of the logical interface.	**detail extensive** none
Broadcast	Broadcast address of the logical interlace.	**detail extensive** none

Table 12: show interfaces Gigabit Ethernet Output Fields *(continued)*

Field Name	Field Description	Level of Output
Generation	Unique number for use by Juniper Networks technical support only.	**detail extensive**

For Gigabit Ethernet IQ PICs, traffic and MAC statistics output varies. Table 13 on page 627 describes the traffic and MAC statistics for two sample interfaces, each of which is sending traffic in packets of 500 bytes (including 478 bytes for the Layer 3 packet, 18 bytes for the Layer 2 VLAN traffic header, and 4 bytes for cyclic redundancy check [CRC] information). In Table 13 on page 627, the **ge-0/3/0** interface is the inbound physical interface, and the **ge-0/0/0** interface is the outbound physical interface. On both interfaces, traffic is carried on logical unit **.50** (VLAN 50).

Table 13: Gigabit Ethernet IQ PIC Traffic and MAC Statistics by Interface Type

Interface Type	Sample Command	Byte and Octet Counts Include	Comments
Inbound physical interface	**show interfaces ge-0/3/0 extensive**	Traffic statistics: Input bytes: 496 bytes per packet, representing the Layer 2 packet MAC statistics: Received octets: 500 bytes per packet, representing the Layer 2 packet + 4 bytes	The additional 4 bytes are for the CRC.
Inbound logical interface	**show interfaces ge-0/3/0.50 extensive**	Traffic statistics: Input bytes: 478 bytes per packet, representing the Layer 3 packet	
Outbound physical interface	**show interfaces ge-0/0/0 extensive**	Traffic statistics: Input bytes: 490 bytes per packet, representing the Layer 3 packet + 12 bytes MAC statistics: Received octets: 478 bytes per packet, representing the Layer 3 packet	For input bytes, the additional 12 bytes includes 6 bytes for the destination MAC address + 4 bytes for VLAN + 2 bytes for the Ethernet type.
Outbound logical interface	**show interfaces ge-0/0/0.50 extensive**	Traffic statistics: Input bytes: 478 bytes per packet, representing the Layer 3 packet	

Sample Output

show interfaces extensive (10-Gigabit Ethernet, LAN PHY Mode, IQ2)

```
user@host> show interfaces xe-5/0/0 extensive
Physical interface: xe-5/0/0, Enabled, Physical link is Up
  Interface index: 177, SNMP ifIndex: 99, Generation: 178
  Link-level type: Ethernet, MTU: 1518, LAN-PHY mode, Speed: 10Gbps, Loopback:
```

```
None, Source filtering: Enabled,
  Flow control: Enabled
  Device flags   : Present Running
  Interface flags: SNMP-Traps Internal: 0x4000
  Link flags     : None
  CoS queues     : 8 supported, 4 maximum usable queues
  Schedulers     : 1024
  Hold-times     : Up 0 ms, Down 0 ms
  Current address: 00:14:f6:b9:f1:f6, Hardware address: 00:14:f6:b9:f1:f6
  Last flapped   : Never
  Statistics last cleared: Never
  Traffic statistics:
   Input  bytes  :           6970332384                    0 bps
   Output bytes  :                    0                    0 bps
   Input  packets:             81050506                    0 pps
   Output packets:                    0                    0 pps
   IPv6 transit statistics:
    Input  bytes  :                   0
    Output bytes  :                   0
    Input  packets:                   0
    Output packets:                   0
  Ingress traffic statistics at Packet Forwarding Engine:
   Input  bytes  :           6970299398                    0 bps
   Input  packets:             81049992                    0 pps
   Drop   bytes  :                    0                    0 bps
   Drop   packets:                    0                    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: 0, Errors: 0, Drops: 0, Collisions: 0, Aged packets: 0,
 FIFO errors: 0, HS link CRC errors: 0,
    MTU errors: 0, Resource errors: 0
  Ingress queues: 4 supported, 4 in use
  Queue counters:       Queued packets  Transmitted packets      Dropped packets

    0 best-effort             81049992             81049992                    0

    1 expedited-fo                   0                    0                    0

    2 assured-forw                   0                    0                    0

    3 network-cont                   0                    0                    0

  Egress queues: 4 supported, 4 in use
  Queue counters:       Queued packets  Transmitted packets      Dropped packets

    0 best-effort                    0                    0                    0

    1 expedited-fo                   0                    0                    0

    2 assured-forw                   0                    0                    0

    3 network-cont                   0                    0                    0

  Active alarms  : None
  Active defects : None
  PCS statistics                      Seconds
    Bit errors                             0
    Errored blocks                         0
```



```
MAC statistics:                      Receive         Transmit
  Total octets                    6970332384                0
  Total packets                     81050506                0
  Unicast packets                   81050000                0
  Broadcast packets                      506                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                81050506
  Input packet rejects                   506
  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: 0, CAM source filters: 0
Packet Forwarding Engine configuration:
  Destination slot: 5
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

  Direction : Input
  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

Logical interface xe-5/0/0.0 (Index 71) (SNMP ifIndex 95) (Generation 195)
  Flags: SNMP-Traps 0x4000 VLAN-Tag [ 0x8100.100 ]  Encapsulation: ENET2
  Egress account overhead: 100
  Ingress account overhead: 90
  Traffic statistics:
   Input  bytes  :                   0
   Output bytes  :                  46
   Input  packets:                   0
   Output packets:                   1
   IPv6 transit statistics:
    Input  bytes  :                  0
    Output bytes  :                  0
    Input  packets:                  0
    Output packets:                  0
  Local statistics:
   Input  bytes  :                   0
   Output bytes  :                  46
   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
      IPv6 transit statistics:
       Input  bytes  :                  0
       Output bytes  :                  0
       Input  packets:                  0
       Output packets:                  0
    Protocol inet, MTU: 1500, Generation: 253, Route table: 0
      Addresses, Flags: Is-Preferred Is-Primary
        Destination: 192.1.1/24, Local: 192.1.1.1, Broadcast: 192.1.1.255,
Generation: 265
    Protocol multiservice, MTU: Unlimited, Generation: 254, Route table: 0
      Flags: None
      Policer: Input: __default_arp_policer__
```

show interfaces extensive (10-Gigabit Ethernet, WAN PHY Mode)

```
user@host> show interfaces xe-1/0/0 extensive
Physical interface: xe-1/0/0, Enabled, Physical link is Up
  Interface index: 141, SNMP ifIndex: 34, Generation: 47
  Link-level type: Ethernet, MTU: 1514, Speed: 10Gbps, Loopback: Disabled
  WAN-PHY mode
  Source filtering: Disabled, Flow control: Enabled
  Device flags   : Present Running
  Interface flags: SNMP-Traps 16384
  Link flags     : None
  CoS queues     : 4 supported
  Hold-times     : Up 0 ms, Down 0 ms
  Current address: 00:05:85:a2:10:9d, Hardware address: 00:05:85:a2:10:9d
  Last flapped   : 2005-07-07 11:22:34 PDT (3d 12:28 ago)
  Statistics last cleared: Never
  Traffic statistics:
   Input  bytes  :                  0                    0 bps
   Output bytes  :                  0                    0 bps
   Input  packets:                  0                    0 pps
   Output packets:                  0                    0 pps
  Input errors:
    Errors: 0, Drops: 0, Framing errors: 0, Runts: 0, Policed discards: 0,
    L3 incompletes: 0, L2 channel errors: 0, L2 mismatch timeouts: 0,
    HS Link CRC errors: 0, HS Link FIFO overflows: 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
  Queue counters:       Queued packets  Transmitted packets     Dropped packets
    0 best-effort                    0                    0                   0
    1 expedited-fo                   0                    0                   0
    2 assured-forw                   0                    0                   0
    3 network-cont                   0                    0                   0
  Active alarms : LOL, LOS, LBL
  Active defects: LOL, LOS, LBL, SEF, AIS-L, AIS-P
  PCS statistics             Seconds       Count
    Bit errors                     0           0
    Errored blocks                 0           0
  MAC statistics:                      Receive         Transmit
    Total octets                             0                0
    Total packets                            0                0
    Unicast packets                          0                0
    Broadcast packets                        0                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: 0, CAM source filters: 0
  PMA PHY:             Seconds        Count  State
    PLL lock                 0            0  OK
    PHY light            63159            1  Light Missing
 WIS section:
   BIP-B1                    0            0
   SEF                  434430       434438  Defect Active
   LOS                  434430            1  Defect Active
   LOF                  434430            1  Defect Active
   ES-S                 434430
   SES-S                434430
   SEFS-S               434430
 WIS line:
   BIP-B2                    0            0
   REI-L                     0            0
   RDI-L                     0            0  OK
   AIS-L                434430            1  Defect Active
   BERR-SF                   0            0  OK
   BERR-SD                   0            0  OK
   ES-L                 434430
   SES-L                434430
   UAS-L                434420
   ES-LFE                    0
   SES-LFE                   0
   UAS-LFE                   0
 WIS path:
   BIP-B3                    0            0
   REI-P                     0            0
   LOP-P                     0            0  OK
   AIS-P                434430            1  Defect Active
   RDI-P                     0            0  OK
   UNEQ-P                    0            0  OK
   PLM-P                     0            0  OK
   ES-P                 434430
   SES-P                434430
   UAS-P                434420
   ES-PFE                    0
   SES-PFE                   0
   UAS-PFE                   0
 Received path trace:
   00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
 Transmitted path trace: orissa so-1/0/0
   6f 72 69 73 73 61 20 73 6f 2d 31 2f 30 2f 30 00   orissa so-1/0/0.
   Packet Forwarding Engine configuration:
```

```
    Destination slot: 1
  CoS information:
    CoS transmit queue          Bandwidth         Buffer     Priority   Limit
                      %               bps     %   bytes
    0 best-effort      95       950000000    95       0         low      none
    3 network-control   5        50000000     5       0         low      none
```

show interfaces extensive (10-Gigabit Ethernet, DWDM OTN PIC)

```
user@host> show interfaces ge-7/0/0 extensive
Physical interface: ge-7/0/0, Enabled, Physical link is Down
  Interface index: 143, SNMP ifIndex: 508, Generation: 208
  Link-level type: Ethernet, MTU: 1514, Speed: 10Gbps, BPDU Error: None,
  MAC-REWRITE Error: None, Loopback: Disabled, Source filtering: Disabled,
  Flow control: Enabled
  Device flags   : Present Running Down
  Interface flags: Hardware-Down SNMP-Traps Internal: 0x4000
  Link flags     : None
  Wavelength     : 1550.12 nm, Frequency: 193.40 THz
  CoS queues     : 8 supported, 8 maximum usable queues
  Hold-times     : Up 0 ms, Down 0 ms
  Current address: 00:05:85:70:2b:72, Hardware address: 00:05:85:70:2b:72
  Last flapped   : 2011-04-20 15:48:54 PDT (18:39:49 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
   IPv6 transit statistics:
    Input  bytes  :                  0
    Output bytes  :                  0
    Input  packets:                  0
    Output packets:                  0
  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: 2, 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                    0                    0                    0

    1 expedited-fo                   0                    0                    0

    2 assured-forw                   0                    0                    0

    3 network-cont
  Queue number:         Mapped forwarding classes
    0                   best-effort
    1                   expedited-forwarding
    2                   assured-forwarding
    3                   network-control
  Active alarms  : LINK
  Active defects : LINK
  MAC statistics:                      Receive         Transmit
    Total octets                             0                0
    Total packets                            0                0
```



```
    Unicast packets                         0                0
    Broadcast packets                       0                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
    Total octets                            0                0
    Total packets                           0                0
    Unicast packets                         0                0
    Broadcast packets                       0                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
  OTN alarms           :   None
  OTN defects          :   None
  OTN FEC Mode       : GFEC
  OTN Rate             : Fixed Stuff Bytes 11.0957Gbps
  OTN Line Loopback : Enabled
  OTN FEC statistics :
    Corrected Errors                                         0
    Corrected Error Ratio (         0 sec average)     0e-0
  OTN FEC alarms:         Seconds        Count  State
    FEC Degrade                 0            0  OK
    FEC Excessive               0            0  OK
  OTN OC:                 Seconds        Count  State
    LOS                         2            1  OK
    LOF                     67164            2  Defect Active
    LOM                     67164           71  Defect Active
    Wavelength Lock             0            0  OK
  OTN OTU:
    AIS                         0            0  OK
    BDI                     65919         4814  Defect Active
    IAE                     67158            1  Defect Active
    TTIM                        7            1  OK
    SF                      67164            2  Defect Active
    SD                      67164            3  Defect Active
    TCA-ES                      0            0  OK
    TCA-SES                     0            0  OK
    TCA-UAS                    80           40  OK
    TCA-BBE                     0            0  OK
    BIP                         0            0  OK
    BBE                         0            0  OK
    ES                          0            0  OK
    SES                         0            0  OK
    UAS                       587            0  OK
    Received DAPI:
    00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
    Received SAPI:
```



```
    00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
    Transmitted DAPI:
    00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
    Transmitted SAPI:
    00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
  OTN Received Overhead Bytes:
    APS/PCC0: 0x02, APS/PCC1: 0x42, APS/PCC2: 0xa2, APS/PCC3: 0x48
    Payload Type: 0x03
  OTN Transmitted Overhead Bytes:
    APS/PCC0: 0x00, APS/PCC1: 0x00, APS/PCC2: 0x00, APS/PCC3: 0x00
    Payload Type: 0x03
  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: 0, CAM source filters: 0
  Packet Forwarding Engine configuration:
    Destination slot: 7
  CoS information:
    Direction : Output
    CoS transmit queue               Bandwidth               Buffer Priority
Limit
                              %            bps     %              usec
    0 best-effort            95     9500000000    95                 0       low
none
    3 network-control         5      500000000     5                 0       low
none
    ...
```

show interfaces extensive (10-Gigabit Ethernet, LAN PHY Mode, Unidirectional Mode)

```
user@host> show interfaces xe-7/0/0 extensive
Physical interface: xe-7/0/0, Enabled, Physical link is Up
  Interface index: 173, SNMP ifIndex: 212, Generation: 174
  Link-level type: Ethernet, MTU: 1514, LAN-PHY mode, Speed: 10Gbps,
Unidirectional: Enabled,
  Loopback: None, Source filtering: Disabled, Flow control: Enabled
  Device flags   : Present Running
...
```

show interfaces extensive (10-Gigabit Ethernet, LAN PHY Mode, Unidirectional Mode, Transmit-Only)

```
user@host> show interfaces xe-7/0/0–tx extensive
Physical interface: xe-7/0/0-tx, Enabled, Physical link is Up
  Interface index: 176, SNMP ifIndex: 137, Generation: 177
  Link-level type: Ethernet, MTU: 1514, LAN-PHY mode, Speed: 10Gbps,
Unidirectional: Tx-Only
  Device flags   : Present Running
  Interface flags: SNMP-Traps Internal: 0x4000
  Link flags     : None
  CoS queues     : 8 supported, 8 maximum usable queues
  Hold-times     : Up 0 ms, Down 0 ms
  Current address: 00:05:85:73:e4:83, Hardware address: 00:05:85:73:e4:83
  Last flapped   : 2007-06-01 09:08:19 PDT (3d 02:31 ago)
  Statistics last cleared: Never
  Traffic statistics:
   Input  bytes  :                    0                    0 bps
```



```
   Output bytes  :      322891152287160          9627472888 bps
   Input  packets:                    0                   0 pps
   Output packets:         328809727380             1225492 pps

...

  Filter statistics:
    Output packet count          328810554250
    Output packet pad count                 0
    Output packet error count               0
...

  Logical interface xe-7/0/0-tx.0 (Index 73) (SNMP ifIndex 138) (Generation 139)

    Flags: SNMP-Traps Encapsulation: ENET2
    Egress account overhead: 100
    Ingress account overhead: 90
    Traffic statistics:
     Input  bytes  :                    0
     Output bytes  :      322891152287160
     Input  packets:                    0
     Output packets:         328809727380
     IPv6 transit statistics:
      Input  bytes  :                   0
      Output bytes  :                   0
      Input  packets:                   0
      Output packets:                   0
    Local statistics:
     Input  bytes  :                    0
     Output bytes  :                    0
     Input  packets:                    0
     Output packets:                    0
    Transit statistics:
     Input  bytes  :                    0                   0 bps
     Output bytes  :      322891152287160          9627472888 bps
     Input  packets:                    0                   0 pps
     Output packets:         328809727380             1225492 pps
     IPv6 transit statistics:
      Input  bytes  :                   0
      Output bytes  :                   0
      Input  packets:                   0
      Output packets:                   0
    Protocol inet, MTU: 1500, Generation: 147, Route table: 0
      Addresses, Flags: Is-Preferred Is-Primary
        Destination: 10.11.12/24, Local: 10.11.12.13, Broadcast: 10.11.12.255,
Generation: 141
    Protocol multiservice, MTU: Unlimited, Generation: 148, Route table: 0
      Flags: None
      Policer: Input: __default_arp_policer__
```

show interfaces extensive (10-Gigabit Ethernet, LAN PHY Mode, Unidirectional Mode, Receive-Only)

```
user@host> show interfaces xe-7/0/0–rx extensive
Physical interface: xe-7/0/0-rx, Enabled, Physical link is Up
  Interface index: 174, SNMP ifIndex: 118, Generation: 175
  Link-level type: Ethernet, MTU: 1514, LAN-PHY mode, Speed: 10Gbps,
Unidirectional: Rx-Only
  Device flags   : Present Running
  Interface flags: SNMP-Traps Internal: 0x4000
  Link flags     : None
  CoS queues     : 8 supported, 8 maximum usable queues
```



```
  Hold-times     : Up 0 ms, Down 0 ms
  Current address: 00:05:85:73:e4:83, Hardware address: 00:05:85:73:e4:83
  Last flapped   : 2007-06-01 09:08:22 PDT (3d 02:31 ago)
  Statistics last cleared: Never
  Traffic statistics:
   Input  bytes  :      322857456303482          9627496104 bps
   Output bytes  :                    0                   0 bps
   Input  packets:         328775413751             1225495 pps
   Output packets:                    0                   0 pps

...

  Filter statistics:
    Input packet count            328775015056
    Input packet rejects                     1
    Input DA rejects                         0

...

  Logical interface xe-7/0/0-rx.0 (Index 72) (SNMP ifIndex 120) (Generation 138)

    Flags: SNMP-Traps Encapsulation: ENET2
    Traffic statistics:
     Input  bytes  :      322857456303482
     Output bytes  :                    0
     Input  packets:         328775413751
     Output packets:                    0
     IPv6 transit statistics:
      Input  bytes  :                   0
      Output bytes  :                   0
      Input  packets:                   0
      Output packets:                   0
    Local statistics:
     Input  bytes  :                    0
     Output bytes  :                    0
     Input  packets:                    0
     Output packets:                    0
    Transit statistics:
     Input  bytes  :      322857456303482          9627496104 bps
     Output bytes  :                    0                   0 bps
     Input  packets:         328775413751             1225495 pps
     Output packets:                    0                   0 pps
     IPv6 transit statistics:
      Input  bytes  :                   0
      Output bytes  :                   0
      Input  packets:                   0
      Output packets:                   0
    Protocol inet, MTU: 1500, Generation: 145, Route table: 0
      Addresses, Flags: Is-Preferred Is-Primary
        Destination: 192.1.1/24, Local: 192.1.1.1, Broadcast: 192.1.1.255,
Generation: 139
    Protocol multiservice, MTU: Unlimited, Generation: 146, Route table: 0
      Flags: None
      Policer: Input: __default_arp_policer__
```


show interfaces (ATM)

Syntax

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

Release Information

Command introduced before Junos OS Release 7.4.

Description

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

Options

at-*fpc/pic/port*—Display standard information about the specified ATM interface.

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

descriptions—(Optional) Display interface description strings.

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

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

statistics—(Optional) Display static interface statistics.

Required Privilege Level

view

List of Sample Output

show interfaces (ATM, IMA Group) on page 652
show interfaces extensive (ATM IMA Group) on page 653
show interfaces (ATM1, SONET Mode) on page 654
show interfaces brief (ATM1, SONET Mode) on page 655
show interfaces detail (ATM1, SONET Mode) on page 655
show interfaces extensive (ATM1, SONET Mode) on page 656
show interfaces (ATM2, SDH Mode) on page 658
show interfaces brief (ATM2, SDH Mode) on page 659
show interfaces detail (ATM2, SDH Mode) on page 660
show interfaces extensive (ATM2, SDH Mode) on page 661
show interfaces (ATM2, SONET Mode) on page 664
show interfaces brief (ATM2, SONET Mode) on page 665
show interfaces detail (ATM2, SONET Mode) on page 666
show interfaces extensive (ATM2, SONET Mode) on page 668

Output Fields

Table 14 on page 637 lists the output fields for the **show interfaces** (ATM) command. Output fields are listed in the approximate order in which they appear.

Table 14: ATM show interfaces Output Fields

Field Name	Field Description	Level of Output
Physical Interface		

Table 14: ATM show interfaces Output Fields *(continued)*

Field Name	Field Description	Level of Output
Physical interface	Name of the physical interface.	All levels
Enabled	State of the interface. Possible values are described in the “Enabled Field” section under *Common Output Fields Description*.	All levels
Description	Configured interface description.	All levels
Interface index	Physical interface's index number, which reflects its initialization sequence.	**detail extensive** none
SNMP ifIndex	SNMP index number for the physical interface.	**detail extensive** none
Generation	Unique number for use by Juniper Networks technical support only.	**detail extensive**
Link-level type	Encapsulation being used on the physical interface: • **ATM-CCC-CELL-RELAY**—ATM cell relay for CCC. • **ATM-CCC-VC-MUX**—ATM virtual circuit (VC) for CCC. • **ATM-CISCO-NLPID**—Cisco-compatible ATM NLPID encapsulation. • **ATM-MIPP-LLC**—ATM MLPPP over ATM Adaptation Layer 5 (AAL5)/logical link control (LLC). • **ATM-NLPID**—ATM NLPID encapsulation. • **ATM-PPP-LLC**—ATM PPP over AAL5/LLC. • **ATM-PPP-VC-MUX**—ATM PPP over raw AAL5. • **ATM-PVC**—ATM permanent virtual circuits. • **ATM-SNAP**—ATM LLC/SNAP encapsulation. • **ATM-TCC-SNAP**—ATM LLC/SNAP for translational cross-connection. • **ATM-TCC-VC-MUX**—ATM VC for translational cross-connection. • **ATM-VC-MUX**—ATM VC multiplexing. • **ETHER-OVER-ATM-LLC**—Ethernet over ATM (LLC/SNAP) encapsulation. • **ETHER-VPLS-OVER-ATM-LLC**—Ethernet VPLS over ATM (bridging) encapsulation.	All levels
MTU	MTU size on the physical interface.	All levels
Clocking	Reference clock source: **Internal** or **External**.	All levels
***framing* Mode**	Framing mode: **SONET** or **SDH**.	All levels
Speed	Speed at which the interface is running as represented by the interface type (for example, **OC3**, **ADSL2+**, and **SHDSL(2-wire)**.	All levels
Loopback	Whether loopback is enabled and the type of loopback (**local** or **remote**).	All levels
Payload scrambler	Whether payload scrambling is enabled.	All levels
Device flags	Information about the physical device. Possible values are described in the “Device Flags” section under *Common Output Fields Description*.	All levels

Table 14: ATM show interfaces Output Fields *(continued)*

Field Name	Field Description	Level of Output
Link flags	Information about the link. Possible values are described in the "Link Flags" section under *Common Output Fields Description*.	All levels
CoS queues	Number of CoS queues configured.	**detail extensive** none
Hold-times	Current interface hold-time up and hold-time down, in milliseconds.	**detail extensive**
Current address	Ethernet MAC address for this interface for Ethernet over ATM encapsulation.	**detail extensive** none
Last flapped	Date, time, and how long ago the interface went from down to up. The format is **Last flapped:** ***year-month-day hour:minute:second timezone*** **(*hour:minute:second* ago)**. For example, **Last flapped: 2002-04-26 10:52:40 PDT (04:33:20 ago)**.	**detail extensive** none
Input Rate	Input rate in bits per second (bps) and packets per second (pps).	None specified
Output Rate	Output rate in bps and pps.	None specified
Statistics last cleared	Time when the statistics for the interface were last set to zero.	**detail extensive**
Traffic statistics	Statistics for traffic on the interface. • **Input bytes**—Number of bytes received on the interface • **Output bytes**—Number of bytes transmitted on the interface. • **Input packets**—Number of packets received on the interface • **Output packets**—Number of packets transmitted on the interface.	**detail extensive**
Input errors	Input errors on the interface whose definitions are as follows: • **Errors**—Sum of the incoming frame aborts and frame check sequence (FCS) errors. • **Drops**—Number of packets dropped by the input queue of the I/O Manager ASIC. If the interface is saturated, this number increments once for every packet that is dropped by the ASIC's random early detection (RED) mechanism. • **Invalid VCs**—Number of cells that arrived for a nonexistent VC. • **Framing errors**—Sum of AAL5 packets that have FCS errors, reassembly timeout errors, and length errors. • **Policed discards**—Number of frames that the incoming packet match code discarded because they were not recognized or not of interest. Usually, this field reports protocols that the Junos OS does not handle. • **L3 incompletes**—Number of incoming packets discarded because they failed Layer 3 (usually IPv4) sanity checks of the header. For example, a frame with less than 20 bytes of available IP header is discarded. • **L2 channel errors**—Number of times the software did not find a valid logical interface for an incoming frame. • **L2 mismatch timeouts**—Number of malformed or short packets that caused the incoming packet handler to discard the frame as unreadable. • **Resource errors**—Sum of transmit drops.	**extensive**

Table 14: ATM show interfaces Output Fields *(continued)*

Field Name	Field Description	Level of Output
Output errors	Output errors on the interface. The following paragraphs explain the counters whose meaning might not be obvious: • **Carrier transitions**—Number of times the interface has gone from **down** to **up**. This number does not normally increment quickly, increasing only when the cable is unplugged, the far-end system is powered down and up, or another problem occurs. If the number of carrier transitions increments quickly, increasing only when the cable is unplugged, the far-end system is powered down and then up, or another problem occurs. If it increments quickly (perhaps once every 10 seconds), the cable, the far-end system, or the PIC or PIM is malfunctioning. • **Errors**—Sum of the outgoing frame aborts and FCS errors. • **Drops**—Number of packets dropped by the output queue of the I/O Manager ASIC. If the interface is saturated, this number increments once for every packet that is dropped by the ASIC's RED mechanism. • **Aged packets**—Number of packets that remained so long in shared packet SDRAM that the system automatically purged them. The value in this field should never increment. If it does, it is most likely a software bug or possibly malfunctioning hardware. • **MTU errors**—Number of packets larger than the MTU threshold. • **Resource errors**—Sum of transmit drops.	**extensive**
Egress queues	Total number of egress queues supported on the specified interface.	**detail extensive**
Queue counters	CoS queue number and its associated user-configured forwarding class name. • **Queued packets**—Number of queued packets. • **Transmitted packets**—Number of transmitted packets. • **Dropped packets**—Number of packets dropped by the ASIC's RED mechanism. NOTE: Physical interface queue counters of ATM2 PICs displayed by the **show interfaces at-*fpc/pic/port* detail** command show the packet forwarding stream statistics associated with the ATM2 ports. Since multiple ports of the ATM2 PICs (except for the ATM2 dual-port OC12) share one packet forwarding stream, the physical interface queue counters reflect the aggregate of ATM2 port statistics.	**detail extensive**
SONET alarms **SONET defects**	SONET media-specific defects that prevent the interface from passing packets. When a defect persists for a certain period, it is promoted to an alarm. Based on the router configuration, an alarm can ring the red or yellow alarm bell on the router or light the red or yellow alarm LED on the craft interface. See these fields for possible alarms and defects: **SONET PHY**, **SONET section**, **SONET line**, and **SONET path**.	**detail extensive** none

Table 14: ATM show interfaces Output Fields *(continued)*

Field Name	Field Description	Level of Output
SONET PHY	Counts of specific SONET errors with detailed information. • **Seconds**—Number of seconds the defect has been active. • **Count**—Number of times that the defect has gone from inactive to active. • **State**—State of the error. State other than **OK** indicates a problem. Subfields are: • **PLL Lock**—Phase-locked loop • **PHY Light**—Loss of optical signal	**extensive**
SONET section	Counts of specific SONET errors with detailed information. • **Seconds**—Number of seconds the defect has been active. • **Count**—Number of times that the defect has gone from inactive to active. • **State**—State of the error. State other than **OK** indicates a problem. Subfields are: • **BIP-B1**—Bit interleaved parity for SONET section overhead • **SEF**—Severely errored framing • **LOL**—Loss of light • **LOF**—Loss of frame • **ES-S**—Errored seconds (section) • **SES-S**—Severely errored seconds (section) • **SEFS-S**—Severely errored framing seconds (section)	**extensive**
SONET line	Active alarms and defects, plus counts of specific SONET errors with detailed information. • **Seconds**—Number of seconds the defect has been active. • **Count**—Number of times that the defect has gone from inactive to active. • **State**—State of the error. State other than **OK** indicates a problem. Subfields are: • **BIP-B2**—Bit interleaved parity for SONET line overhead • **REI-L**—Remote error indication (near-end line) • **RDI-L**—Remote defect indication (near-end line) • **AIS-L**—Alarm indication signal (near-end line) • **BERR-SF**—Bit error rate fault signal failure • **BERR-SD**—Bit error rate defect signal degradation • **ES-L**—Errored seconds (near-end line) • **SES-L**—Severely errored seconds (near-end line) • **UAS-L**—Unavailable seconds (near-end line) • **ES-LFE**—Errored seconds (far-end line) • **SES-LFE**—Severely errored seconds (far-end line) • **UAS-LFE**—Unavailable seconds (far-end line)	**extensive**

Table 14: ATM show interfaces Output Fields *(continued)*

Field Name	Field Description	Level of Output
SONET path	Active alarms and defects, plus counts of specific SONET errors with detailed information. • **Seconds**—Number of seconds the defect has been active. • **Count**—Number of times that the defect has gone from inactive to active. • **State**—State of the error. State other than **OK** indicates a problem. Subfields are: • **BIP-B3**—Bit interleaved parity for SONET section overhead • **REI-P**—Remote error indication • **LOP-P**—Loss of pointer (path) • **AIS-P**—Path alarm indication signal • **RDI-P**—Path remote defect indication • **UNEQ-P**—Path unequipped • **PLM-P**—Path payload (signal) label mismatch • **ES-P**—Errored seconds (near-end STS path) • **SES-P**—Severely errored seconds (near-end STS path) • **UAS-P**—Unavailable seconds (near-end STS path) • **ES-PFE**—Errored seconds (far-end STS path) • **SES-PFE**—Severely errored seconds (far-end STS path) • **UAS-PFE**—Unavailable seconds (far-end STS path)	**extensive**
Received SONET overhead **Transmitted SONET overhead**	Values of the received and transmitted SONET overhead: • **C2**—Signal label. Allocated to identify the construction and content of the STS-level SPE and for PDI-P. • **F1**—Section user channel byte. This byte is set aside for the purposes of users. • **K1** and **K2**—These bytes are allocated for APS signaling for the protection of the multiplex section. • **J0**—Section trace. This byte is defined for STS-1 number 1 of an STS-*N* signal. Used to transmit a 1-byte fixed-length string or a 16-byte message so that a receiving terminal in a section can verify its continued connection to the intended transmitter. • **S1**—Synchronization status. The S1 byte is located in the first STS-1 of an STS-*N*. • **Z3** and **Z4**—Allocated for future use.	**extensive**
SDH alarms **SDH defects**	SDH media-specific defects that can prevent the interface from passing packets. When a defect persists for a certain period, it is promoted to an alarm. Based on the router configuration, an alarm can ring the red or yellow alarm bell on the router or light the red or yellow alarm LED on the craft interface. See these fields for possible alarms and defects: **SDH PHY**, **SDH regenerator section**, **SDH multiplex section**, and **SDH path**.	All levels

Table 14: ATM show interfaces Output Fields *(continued)*

Field Name	Field Description	Level of Output
SDH PHY	Active alarms and defects, plus counts of specific SDH errors with detailed information. • **Seconds**—Number of seconds the defect has been active. • **Count**—Number of times that the defect has gone from inactive to active. • **State**—State of the error. State other than **OK** indicates a problem. Subfields are: • **PLL Lock**—Phase-locked loop • **PHY Light**—Loss of optical signal	**extensive**
SDH regenerator section	Active alarms and defects, plus counts of specific SDH errors with detailed information. • **Seconds**—Number of seconds the defect has been active. • **Count**—Number of times that the defect has gone from inactive to active. • **State**—State of the error. State other than **OK** indicates a problem. Subfields are: • **RS-BIP8**—24-bit BIP for multiplex section overhead (B2 bytes) • **OOF**—Out of frame • **LOS**—Loss of signal • **LOF**—Loss of frame • **RS-ES**—Errored seconds (near-end regenerator section) • **RS-SES**—Severely errored seconds (near-end regenerator section) • **RS-SEFS**—Severely errored framing seconds (regenerator section)	**extensive**
SDH multiplex section	Active alarms and defects, plus counts of specific SDH errors with detailed information. • **Seconds**—Number of seconds the defect has been active. • **Count**—Number of times that the defect has gone from inactive to active. • **State**—State of the error. State other than **OK** indicates a problem. Subfields are: • **MS-BIP24**—8-bit BIP for high-order path overhead (B3 byte) • **MS-FEBE**—Far-end block error (multiplex section) • **MS-FERF**—Far-end remote fail (multiplex section) • **MS-AIS**—Alarm indication signal (multiplex section) • **BERR-SF**—Bit error rate fault (signal failure) • **BERR-SD**—Bit error rate defect (signal degradation) • **MS-ES**—Errored seconds (near-end multiplex section) • **MS-SES**—Severely errored seconds (near-end multiplex section) • **MS-UAS**—Unavailable seconds (near-end multiplex section) • **MS-ES-FE**—Errored seconds (far-end multiplex section) • **MS-SES-FE**—Severely errored seconds (far-end multiplex section) • **MS-UAS-FE**—Unavailable seconds (far-end multiplex section)	**extensive**

Table 14: ATM show interfaces Output Fields *(continued)*

Field Name	Field Description	Level of Output
SDH path	Active alarms and defects, plus counts of specific SDH errors with detailed information. • **Seconds**—Number of seconds the defect has been active. • **Count**—Number of times that the defect has gone from inactive to active. • **State**—State of the error. State other than **OK** indicates a problem. Subfields are: • **HP-BIP8**—8-bit BIP for regenerator section overhead (B1 byte) • **HP-FEBE**—Far-end block error (high-order path) • **HP-LOP**—Loss of pointer (high-order path) • **HP-AIS**—High-order-path alarm indication signal • **HP-FERF**—Far-end remote fail (high-order path) • **HP-UNEQ**—Unequipped (high-order path) • **HP-PLM**—Payload label mismatch (high-order path) • **HP-ES**—Errored seconds (near-end high-order path) • **HP-SES**—Severely errored seconds (near-end high-order path) • **HP-UAS**—Unavailable seconds (near-end high-order path) • **HP-ES-FE**—Errored seconds (far-end high-order path) • **HP-SES-FE**—Severely errored seconds (far-end high-order path) • **HP-UAS-FE**—Unavailable seconds (far-end high-order path)	**extensive**
Received SDH overhead **Transmitted SDH overhead**	Values of the received and transmitted SONET overhead: • **C2**—Signal label. This byte is allocated to identify the construction and content of the STS-level SPE and for PDI-P. • **F1**—Section user channel byte. This byte is set aside for the purposes of users. • **K1** and **K2**—These bytes are allocated for APS signaling for the protection of the multiplex section. • **J0**—Section trace. This byte is defined for STS-1 number 1 of an STS-*N* signal. This bye is used to transmit a 1-byte fixed-length string or a 16-byte message so that a receiving terminal in a section can verify its continued connection to the intended transmitter. • **S1**—Synchronization status. The S1 byte is located in the first STS-1 of an STS-*N*. • **Z3** and **Z4**—These bytes are allocated for future use.	**extensive**
Received path trace **Transmitted path trace**	SONET/SDH interfaces allow path trace bytes to be sent inband across the SONET/SDH link. Juniper Networks and other router manufacturers use these bytes to help diagnose misconfigurations and network errors by setting the transmitted path trace message so that it contains the system hostname and name of the physical interface. The received path trace value is the message received from the router at the other end of the fiber. The transmitted path trace value is the message that this router transmits.	**extensive**

Table 14: ATM show interfaces Output Fields *(continued)*

Field Name	Field Description	Level of Output
ATM Status	ATM state information: • **HCS State**—Status of the header check sequence. ATM uses the HCS field in the cell header in the cell delineation process to frame ATM cell boundaries. The HCS is an FCS-8 calculation over the first four octets of the ATM cell header. • **LOC**—Current loss of cell (LOC) delineation state. **OK** means that no LOC is currently asserted.	**extensive**

Table 14: ATM show interfaces Output Fields *(continued)*

Field Name	Field Description	Level of Output
ATM Statistics	ATM statistics for the interface: • **Uncorrectable HCS errors**—Number of cells dropped because the cell delineation failed. These errors most likely indicate that a SONET/SDH layer problem has occurred. • **Correctable HCS errors**—Number of correctable HCS errors that occurred. The cell delineation process can recover from these errors and locate the ATM cell boundary, although the framing process is not quite stable. The ATM cell is not dropped. This counter increases when the cell delineation process changes its state from **present** to **sync** (for example, when a cable is plugged into the interface). The following error statistics are from the framer: • **Tx cell FIFO overruns**—Number of overruns in the transmit FIFO. • **Rx cell FIFO overruns**—Number of overruns in the receive FIFO. • **Rx cell FIFO underruns**—Number of underruns in the receive FIFO. • **Input cell count**—Number of ATM cells received by the interface (not including idle cells). • **Output cell count**—Number of ATM cells transmitted by the interface (including idle cells). • **Output idle cell count**—Number of idle cells sent by the port. When ATM has nothing to send, it sends idle cells to fill the time slot. • **Output VC queue drops**—Number of packets dropped by a port on the PIC. Packets are dropped because of queue limits on the VCs. The following error statistics are from the SAR: • **Input no buffers**—Number of AAL5 packets dropped because no channel blocks or buffers were available to handle them. • **Input length errors**—Number of AAL5 packets dropped because their length was incorrect. Usually, these errors occur because a cell has been corrupted or lost, or because the length field was corrupted. They can also mean the AAL5 length field was zero. • **Input timeouts**—Number of AAL5 packets dropped because of a reassembly timeout. • **Input invalid VCs**—Number of AAL5 packets dropped because the header was unrecognized (because the VC was not correct or not configured). • **Input bad CRCs**—Number of AAL5 packets dropped because of frame check sequence errors. • **Input OAM cell no buffers**—Number of received OAM cells or raw cells dropped because no buffers were available to handle them. • **L2 circuit out-of-sequence packets**—(Layer 2 AAL5 mode) Number of AAL5 packets that are out of sequential order. • **Denied packets count**—The number of packets dropped due to VLAN priority deny packets or due to an error forwarding configuration that might cause a negative frame length, that is, the stripping size is larger than the packet size.	**extensive**
Packet Forwarding Engine configuration	Information about the configuration of the Packet Forwarding Engine: • **Destination slot**—FPC slot number.	**extensive**

Table 14: ATM show interfaces Output Fields *(continued)*

Field Name	Field Description	Level of Output
CoS information	Information about the CoS queue for the physical interface. • **CoS transmit queue**—Queue number and its associated user-configured forwarding class name. • **Bandwidth %**—Percentage of bandwidth allocated to the queue. • **Bandwidth bps**—Bandwidth allocated to the queue (in bps). • **Buffer %**—Percentage of buffer space allocated to the queue. • **Buffer usec**—Amount of buffer space allocated to the queue, in microseconds. This value is nonzero only if the buffer size is configured in terms of time. • **Priority**—Queue priority: **low** or **high**. • **Limit**—Displayed if rate limiting is configured for the queue. Possible values are **none** and **exact**. If **exact** is configured, the queue transmits only up to the configured bandwidth, even if excess bandwidth is available. If **none** is configured, the queue transmits beyond the configured bandwidth if bandwidth is available.	**extensive**

Table 14: ATM show interfaces Output Fields *(continued)*

Field Name	Field Description	Level of Output

VPI	(ATM2) Virtual path identifier information:
	• **Flags**—VPI flags can be one or more of the following:
	  ◦ **Active** (virtual path is up)
	  ◦ **OAM** (operation and maintenance is enabled)
	  ◦ **Shaping** (shaping is configured)
	• **CBR, Peak**
	• **OAM, Period**—Interval at which OAM F4 loopback cells are sent.
	• **Up count**—Number of F4 OAM cells required to consider the virtual path up; the range is **1** through **255**.
	• **Down count**—Number of F4 OAM cells required to consider the virtual path down; the range is **1** through **255**.
	• **Total down time**—Total number of seconds the VPI has been down since it was opened, using the format **Total down time:** ***hh:mm:ss*** or **Never.**
	• **Last down**—Time of last **Down** transition, using the format **Last down:** ***hh:mm:ss*** **ago** or **Never**.
	• **OAM F4 cell statistics**—(Nonpromiscuous mode) OAM F4 statistics:
	  ◦ **Total received**—Number of OAM F4 cells received.
	  ◦ **Total sent**—Number of OAM F4 cells sent.
	  ◦ **Loopback received**—Number of OAM F4 loopback cells received.
	  ◦ **Loopback sent**—Number of OAM F4 loopback cells sent.
	  ◦ **Last received**—Time at which the last OAM F4 cell was received.
	  ◦ **Last sent**—Time at which the last OAM F4 cell was sent.
	  ◦ **RDI received**—Number of OAM F4 cells received with the remote defect indication bit set.
	  ◦ **RDI sent**—Number of OAM F4 cells sent with the RDI bit set.
	  ◦ **AIS received**—Number of OAM F4 cells received with the alarm indication signal bit set.
	  ◦ **AIS sent**—Number of OAM F4 cells sent with the AIS bit set.
	Traffic statistics:
	• **Input bytes**—Number of bytes received on the VPI.
	• **Output bytes**—Number of bytes transmitted on the VPI.
	• **Input packets**—Number of packets received on the VPI.
	• **Output packets**—Number of packets transmitted on the VPI.
		detail extensive none

Logical Interface		
Logical interface	Name of the logical interface.	All levels
Index	Logical interface index number, which reflects its initialization sequence.	**detail extensive** none
SNMP ifIndex	Logical interface SNMP interface index number.	**detail extensive** none
Generation	Unique number for use by Juniper Networks technical support only.	**detail extensive**

Table 14: ATM show interfaces Output Fields *(continued)*

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

Table 14: ATM show interfaces Output Fields *(continued)*

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

VCI	Virtual circuit identifier number and information:
	• **Flags**—VCI flags:
	  ◦ **Active**—VCI is up and in working condition.
	  ◦ **CCC down**—VCI CCC is not in working condition.
	  ◦ **Closed**—VCI is closed because the user disabled the logical or physical interface from the CLI.
	  ◦ **Configured**—VCI is configured.
	  ◦ **Down**—VCI is not in working condition. The VCI might have alarms, defects, F5 AIS/RDI, or no response to OAM loopback cells.
	  ◦ **ILMI**—VCI is up and in working condition.
	  ◦ **OAM**—OAM loopback is enabled.
	  ◦ **Multicast**—VCI is a multicast VCI or DLCI.
	  ◦ **Multipoint destination**—VCI is configured as a multipoint destination.
	  ◦ **None**—No VCI flags.
	  ◦ **Passive-OAM**—Passive OAM is enabled.
	  ◦ **Shaping**—Shaping is enabled.
	  ◦ **Sustained**—Shaping rate is set to **Sustained**.
	  ◦ **Unconfigured**—VCI is not configured.
	• **Total down time**—Total number of seconds the VCI has been down, using the format **Total down time:** ***hh:mm:ss*** or**Never**.
	• **Last down**—Time of last **Down** transition, using the format **Last down:** ***hh:mm:ss***.
	• **EPD threshold**—(ATM2 only) Threshold at which a packet is dropped when the queue size (in number of cells) exceeds the early packet-discard (EPD) value.
		All levels

Table 14: ATM show interfaces Output Fields *(continued)*

Field Name	Field Description	Level of Output

VCI (continued)	• **Transmit weight cells**—(ATM2 only) Amount of bandwidth assigned to this queue.
	• **ATM per-VC transmit statistics**:
	  ◦ **Tail queue packet drops**—Number of packets dropped because of bandwidth constraints. This value indicates that packets are queued to send out at a rate faster than allowed.
	• **OAM F4 cell statistics**—(Nonpromiscuous mode) OAM F4 statistics:
	  ◦ **Total received**—Number of OAM F4 cells received.
	  ◦ **Total sent**—Number of OAM F4 cells sent.
	  ◦ **Loopback received**—Number of OAM F4 loopback cells received.
	  ◦ **Loopback sent**—Number of OAM F4 loopback cells sent.
	  ◦ **Last received**—Time at which the last OAM F4 cell was received.
	  ◦ **Last sent**—Time at which the last OAM F4 cell was sent.
	  ◦ **RDI received**—Number of OAM F4 cells received with the remote defect indication bit set.
	  ◦ **RDI sent**—Number of OAM F4 cells sent with the RDI bit set.
	  ◦ **AIS received**—Number of OAM F4 cells received with the alarm indication signal bit set.
	  ◦ **AIS sent**—Number of OAM F4 cells sent with the AIS bit set.
	• **Traffic statistics**—Number and rate of bytes and packets received and transmitted on the physical interface.
	  ◦ **Input bytes**—Number of bytes received on the interface.
	  ◦ **Output bytes**—Number of bytes transmitted on the interface.
	  ◦ **Input packets**—Number of packets received on the interface
	  ◦ **Output packets**—Number of packets transmitted on the interface.
		All levels

IMA group properties	• **Version**—The specified IMA specification version, either IMA 1.0 or IMA 1.1.
	• **Frame length**—The specified frame size, which can be 32, 64, 128, or 256.
	• **Differential delay**—Maximum differential delay among links in milliseconds.
	• **Symmetry**—Either Common Transmit Clock or Independent Transmit Clock timing mode.
	• **Transmit clock**—The specified IMA clock mode, either common or independent.
	• **Minimum links**—The number of minimum active links specified in both transmit and receive directions.
	  ◦ **Transmit**—The per-PIC limit on the number of minimum active links in the transmit direction.
	  ◦ **Receive**—The per-PIC limit on the number of minimum active links in the receive direction.
	• **Frame synchronization**—The specified IMA frame synchronization state transition variables (Alpha, Beta, and Gamma) and their specified values.
	  ◦ **Alpha**—The number of consecutive invalid ICP cells for IFSM.
	  ◦ **Beta**—The number of consecutive errored ICP cells for IFSM.
	  ◦ **Gamma**—The number of consecutive valid ICP cells for IFSM.
	• **Links**—The number of IMA links assigned to the IMA group.
		detail extensive none

Table 14: ATM show interfaces Output Fields *(continued)*

Field Name	Field Description	Level of Output
IMA group alarms	• **Start-up-FE**—Far-end group alarm status • **Config-Aborted**—Near-end configuration aborted group alarm status • **Config-Aborted-FE**—Far-end configuration aborted group alarm status • **Insufficient-Links**—Near-end insufficient links group alarm status • **Insufficient-Links-FE**—Far-end insufficient links group alarm status • **Blocked-FE**—Far-end blocked group alarm status • **GR-Timing-Mismatch**—Group timing mismatch alarm status	**detail extensive** none
IMA group defects	• **Start-up-FE**—Far-end group defect status • **Config-Aborted**—Near-end configuration aborted group defect status • **Config-Aborted-FE**—Far-end configuration aborted group defect status • **Insufficient-Links**—Near-end insufficient links group defect status • **Insufficient-Links-FE**—Far-end insufficient links group defect status • **Blocked-FE**—Far-end blocked group defect status • **GR-Timing-Mismatch**—Group timing mismatch defect status	**detail extensive** none
IMA Group state	Near-end and far-end group status	**detail extensive** none
IMA group media	IMA group media status, including seconds, count and state for the following media parameters: • **FC** • **FC-FE** • **Addr-Mismatch** • **Running** • **UAS**	**detail extensive** none

Sample Output

show interfaces (ATM, IMA Group)

```
user@host> show interfaces at-1/0/0
Physical interface: at-1/0/0, Enabled, Physical link is Up
  IMA group properties:
    Version              : 1.1
    Frame length         : 128
    Differential delay   : 25 milliseconds
    Symmetry             : Symmetrical Configuration and Operation
    Transmit clock       : Common
    Minimum links        : Transmit: 1, Receive: 1
    Frame synchronization: Alpha: 2, Beta: 2, Gamma: 1
    Links                : None
  IMA group alarms  : Start-up-FE Config-Aborted Config-Aborted-FE
Insufficient-Links Insufficient-Links-FE Blocked-FE GR-Timing-Mismatch
  IMA group defects : Start-up-FE Config-Aborted Config-Aborted-FE
Insufficient-Links Insufficient-Links-FE Blocked-FE GR-Timing-Mismatch
  IMA Group state:
    Near end : Start up
    Far  end : Start up
  IMA group media:         Seconds        Count  State
```

```
FC                                          0
FC-FE                                       0
Addr-Mismatch                               0
Running                       0
UAS                           0
```

show interfaces extensive (ATM IMA Group)

```
user@host> show interfaces at-0/0/10 extensive
Physical interface: at-0/0/10, Enabled, Physical link is Up
   Interface index: 178, SNMP ifIndex: 540, Generation: 531
   Link-level type: ATM-PVC, MTU: 2048, Speed: Unspecified, Loopback: None, Payload
 scrambler: Enabled
   Device flags   : Present Running
   Link flags     : None
   CoS queues     : 8 supported, 4 maximum usable queues
   Hold-times     : Up 0 ms, Down 0 ms
   Current address: 84:18:88:c0:33:0a
   Last flapped   : 2012-03-16 16:49:15 PDT (2d 07:12 ago)
   Statistics last cleared: 2012-03-16 16:56:58 PDT (2d 07:05 ago)
   Traffic statistics:
    Input  bytes  :                    0                    0 bps
    Output bytes  :                    0                    0 bps
    Input  packets:                    0                    0 pps
    Output packets:                    0                    0 pps
    IPv6 transit statistics:
     Input  bytes  :                   0
     Output bytes  :                   0
     Input  packets:                   0
     Output packets:                   0
   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
   IMA group properties:
     Version              : 1.1
     Frame length         : 128
     Differential delay   : 25 milliseconds
     Symmetry             : Symmetrical Configuration and Operation
     Transmit clock       : Common
     Minimum links        : Transmit: 1, Receive: 1
     Frame synchronization: Alpha: 2, Beta: 2, Gamma: 1
     Link #1              : t1-0/0/4             up
   IMA Group alarms   : None
   IMA Group defects  : None

   IMA Group state:
     Near end : Operational
     Far  end : Operational
   IMA group media:         Seconds        Count  State
     FC                                      0
     FC-FE                                   0
     Addr-Mismatch                           0
     Running                  198306
     UAS                           0
   ATM status:
     HCS state:     Sync
     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: 0, Output cell count: 0, Output
 idle cell count: 0,
    Output VC queue drops: 0, Input 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: 0
    VPI 2
      Flags: Active
      Total down time: 0 sec, Last down: Never
        Traffic statistics:
         Input  bytes  :                      0
         Output bytes  :                      0
         Input  packets:                      0
         Output packets:                      0

  Logical interface at-0/0/10.602 (Index 71) (SNMP ifIndex 1057) (Generation
17226)
     Flags: Point-To-Point SNMP-Traps CCC-Down 0x0 Encapsulation:
ATM-CCC-Cell-Relay
     L2 circuit cell bundle size: 1, bundle timeout: 125 usec, timeout count: 0
     L2 circuit out-of-sequence count: 0, denied packets count: 0
```

show interfaces (ATM1, SONET Mode)

```
user@host> show interfaces at-1/0/0
Physical interface: at-1/0/0, Enabled, Physical link is Up
  Interface index: 300, SNMP ifIndex: 194
  Description: to allspice at-1/0/0
  Link-level type: ATM-PVC, MTU: 4482, Clocking: Internal, SONET mode,
  Speed: OC3, Loopback: None, Payload scrambler: Enabled
  Device flags   : Present Running
  Link flags     : None
  CoS queues     : 4 supported, 4 maximum usable queues
  Current address: 00:05:85:02:38:7e
  Last flapped   : 2006-02-24 14:28:12 PST (6d 01:51 ago)
  Input rate     : 0 bps (0 pps)
  Output rate    : 0 bps (0 pps)
  SONET alarms   : None
  SONET defects  : None

  Logical interface at-1/0/0.0 (Index 64) (SNMP ifIndex 204)
    Flags: Point-To-Point SNMP-Traps Encapsulation: ATM-SNAP
    Input packets : 0
    Output packets: 0
    Protocol inet, MTU: 4470
      Flags: None
      Addresses, Flags: Is-Preferred Is-Primary
        Destination: 192.168.220.24/30, Local: 192.168.220.26,
        Broadcast: 192.168.220.27
    Protocol iso, MTU: 4470
      Flags: None
    VCI 0.128
      Flags: Active
      Total down time: 0 sec, Last down: Never
        Input packets : 0
        Output packets: 0
```

show interfaces brief (ATM1, SONET Mode)

```
user@host> show interfaces at-1/0/0 brief
Physical interface: at-1/0/0, Enabled, Physical link is Up
  Description: to allspice at-1/0/0
  Link-level type: ATM-PVC, MTU: 4482, Clocking: Internal, SONET mode,
  Speed: OC3, Loopback: None, Payload scrambler: Enabled
  Device flags   : Present Running
  Link flags     : None

  Logical interface at-1/0/0.0
    Flags: Point-To-Point SNMP-Traps Encapsulation: ATM-SNAP
    inet  192.168.220.26/30
    iso
    VCI 0.128
      Flags: Active
      Total down time: 0 sec, Last down: Never
```

show interfaces detail (ATM1, SONET Mode)

```
user@host> show interfaces at-1/0/0 detail
Physical interface: at-1/0/0, Enabled, Physical link is Up
  Interface index: 300, SNMP ifIndex: 194, Generation: 183
  Description: to allspice at-1/0/0
  Link-level type: ATM-PVC, MTU: 4482, Clocking: Internal, SONET mode,
  Speed: OC3, Loopback: None, Payload scrambler: Enabled
  Device flags   : Present Running
  Link flags     : None
  CoS queues     : 4 supported, 4 maximum usable queues
  Hold-times     : Up 0 ms, Down 0 ms
  Current address: 00:05:85:02:38:7e
  Last flapped   : 2006-02-24 14:28:12 PST (6d 01:55 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
  Egress queues: 4 supported, 4 in use
  Queue counters:       Queued packets  Transmitted packets      Dropped packets

    0 best-effort                    0                    0                    0

    1 expedited-fo                   0                    0                    0

    2 assured-forw                   0                    0                    0

    3 network-cont                   0                    0                    0

  SONET alarms   : None
  SONET defects  : None

  Logical interface at-1/0/0.0 (Index 64) (SNMP ifIndex 204) (Generation 5)
    Flags: Point-To-Point SNMP-Traps Encapsulation: ATM-SNAP
    Traffic statistics:
     Input  bytes  :                    0
     Output bytes  :                    0
     Input  packets:                    0
     Output packets:                    0
    Local statistics:
     Input  bytes  :                    0
```

```
     Output bytes  :                    0
     Input  packets:                    0
     Output packets:                    0
    Transit statistics:
     Input  bytes  :                    0                    0 bps
     Output bytes  :                    0                    0 bps
     Input  packets:                    0                    0 pps
     Output packets:                    0                    0 pps
    Protocol inet, MTU: 4470, Generation: 13, Route table: 0
      Flags: None
      Addresses, Flags: Is-Preferred Is-Primary
        Destination: 192.168.220.24/30, Local: 192.168.220.26,
        Broadcast: 192.168.220.27, Generation: 14
    Protocol iso, MTU: 4470, Generation: 14, Route table: 0
      Flags: None
    VCI 0.128
      Flags: Active
      Total down time: 0 sec, Last down: Never
      ATM per-VC transmit statistics:
       Tail queue packet drops: 0
      Traffic statistics:
       Input  bytes  :                    0
       Output bytes  :                    0
       Input  packets:                    0
       Output packets:                    0
```

show interfaces extensive (ATM1, SONET Mode)

```
user@host> show interfaces at-1/0/0 extensive
Physical interface: at-1/0/0, Enabled, Physical link is Up
  Interface index: 300, SNMP ifIndex: 194, Generation: 183
  Description: to allspice at-1/0/0
  Link-level type: ATM-PVC, MTU: 4482, Clocking: Internal, SONET mode,
  Speed: OC3, Loopback: None, Payload scrambler: Enabled
  Device flags   : Present Running
  Link flags     : None
  CoS queues     : 4 supported, 4 maximum usable queues
  Hold-times     : Up 0 ms, Down 0 ms
  Current address: 00:05:85:02:38:7e
  Last flapped   : 2006-02-24 14:28:12 PST (6d 01:56 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 channel errors: 0, L2 mismatch timeouts: 0,
    Resource errors: 0
  Output errors:
    Carrier transitions: 1, Errors: 0, Drops: 0, Aged packets: 0, MTU errors: 0,

    Resource errors: 0
  Egress queues: 4 supported, 4 in use
  Queue counters:       Queued packets  Transmitted packets      Dropped packets

    0 best-effort                    0                    0                    0

    1 expedited-fo                   0                    0                    0
```



```
  2 assured-forw                  0                     0                     0

  3 network-cont                  0                     0                     0

SONET alarms   : None
SONET defects  : None
SONET PHY:            Seconds        Count  State
  PLL Lock                  0            0  OK
  PHY Light                 0            0  OK
SONET section:
  BIP-B1                    0            0
  SEF                       0            0  OK
  LOS                       0            0  OK
  LOF                       0            0  OK
  ES-S                      0
  SES-S                     0
  SEFS-S                    0
SONET line:
  BIP-B2                    0            0
  REI-L                     0            0
  RDI-L                     0            0  OK
  AIS-L                     0            0  OK
  BERR-SF                   0            0  OK
  BERR-SD                   0            0  OK
  ES-L                      0
  SES-L                     0
  UAS-L                     0
  ES-LFE                    0
  SES-LFE                   0
  UAS-LFE                   0
SONET path:
  BIP-B3                    0            0
  REI-P                     0            0
  LOP-P                     0            0  OK
  AIS-P                     0            0  OK
  RDI-P                     0            0  OK
  UNEQ-P                    1            1  OK
  PLM-P                     0            0  OK
  ES-P                      1
  SES-P                     1
  UAS-P                     0
  ES-PFE                    0
  SES-PFE                   0
  UAS-PFE                   0
Received SONET overhead:
  F1      : 0x00, J0      : 0x00, K1      : 0x00, K2      : 0x00
  S1      : 0x00, C2      : 0x13, C2(cmp) : 0x13, F2      : 0x00
  Z3      : 0x00, Z4      : 0x00, S1(cmp) : 0x00
Transmitted SONET overhead:
  F1      : 0x00, J0      : 0x01, K1      : 0x00, K2      : 0x00
  S1      : 0x00, C2      : 0x13, F2      : 0x00, Z3      : 0x00
  Z4      : 0x00
ATM status:
  HCS state:     Sync
  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: 0, Output cell count: 0,
  Output idle cell count: 0, Output VC queue drops: 0, Input 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
  CoS information:
    CoS transmit queue         Bandwidth         Buffer      Priority    Limit
                        %            bps     %     usec
    0 best-effort      95      147744000    95        0           low    none
    3 network-control   5        7776000     5        0           low    none

  Logical interface at-1/0/0.0 (Index 64) (SNMP ifIndex 204) (Generation 5)
    Flags: Point-To-Point SNMP-Traps Encapsulation: ATM-SNAP
    Traffic statistics:
     Input  bytes  :                  0
     Output bytes  :                  0
     Input  packets:                  0
     Output packets:                  0
    Local statistics:
     Input  bytes  :                  0
     Output bytes  :                  0
     Input  packets:                  0
     Output packets:                  0
    Transit statistics:
     Input  bytes  :                  0                    0 bps
     Output bytes  :                  0                    0 bps
     Input  packets:                  0                    0 pps
     Output packets:                  0                    0 pps
    Protocol inet, MTU: 4470, Generation: 13, Route table: 0
      Flags: None
      Addresses, Flags: Is-Preferred Is-Primary
        Destination: 192.168.220.24/30, Local: 192.168.220.26,
        Broadcast: 192.168.220.27, Generation: 14
    Protocol iso, MTU: 4470, Generation: 14, Route table: 0
      Flags: None
    VCI 0.128
      Flags: Active
      Total down time: 0 sec, Last down: Never
      ATM per-VC transmit statistics:
       Tail queue packet drops: 0
      Traffic statistics:
       Input  bytes  :                  0
       Output bytes  :                  0
       Input  packets:                  0
       Output packets:                  0
```

show interfaces (ATM2, SDH Mode)

```
user@host> show interfaces at-0/2/1
Physical interface: at-0/2/1, Enabled, Physical link is Up
  Interface index: 154, SNMP ifIndex: 42
  Link-level type: ATM-PVC, MTU: 4482, Clocking: Internal, SDH mode, Speed: OC3,

  Loopback: None, Payload scrambler: Enabled
  Device flags    : Present Running
  Link flags      : None
  CoS queues      : 4 supported, 4 maximum usable queues
  Current address: 00:05:85:8f:30:3f
  Last flapped    : 2006-03-24 13:29:58 PST (00:04:48 ago)
  Input rate      : 0 bps (0 pps)
  Output rate     : 0 bps (0 pps)
  SDH   alarms    : None
```



```
SDH   defects  : None
  VPI 0
    Flags: Active
    Total down time: 0 sec, Last down: Never
Traffic statistics:
       Input  packets:                    0
       Output packets:                    0

Logical interface at-0/2/1.0 (Index 75) (SNMP ifIndex 51)
  Flags: Point-To-Point SNMP-Traps 0x4000 Encapsulation: ATM-SNAP
  Input packets : 0
  Output packets: 0
  Protocol inet, MTU: 4470
    Flags: None
    Addresses, Flags: Is-Preferred Is-Primary
      Destination: 10.0.12.6, Local: 10.0.12.5
  Protocol iso, MTU: 4470
    Flags: None
  VCI 0.128
    Flags: Active
    Total down time: 0 sec, Last down: Never
    EPD threshold: 2129, Transmit weight cells: 0
      Input packets : 0
      Output packets: 0

Logical interface at-0/2/1.32767 (Index 76) (SNMP ifIndex 50)
  Flags: Point-To-Multipoint No-Multicast SNMP-Traps 0x4000
  Encapsulation: ATM-VCMUX
  Input packets : 0
  Output packets: 0
  VCI 0.4
    Flags: Active
    Total down time: 0 sec, Last down: Never
    EPD threshold: 0, Transmit weight cells: 0
      Input packets : 0
      Output packets: 0
```

show interfaces brief (ATM2, SDH Mode)

```
user@host> show interfaces at-0/2/1 brief
Physical interface: at-0/2/1, Enabled, Physical link is Up
  Link-level type: ATM-PVC, MTU: 4482, Clocking: Internal, SDH mode,
  Speed: OC3, Loopback: None, Payload scrambler: Enabled
  Device flags   : Present Running
  Link flags     : None
  Logical interface at-0/2/1.0
    Flags: Point-To-Point SNMP-Traps 0x4000 Encapsulation: ATM-SNAP
    inet  10.0.12.5        --> 10.0.12.6
    iso
    VCI 0.128
      Flags: Active
      Total down time: 0 sec, Last down: Never
      EPD threshold: 2129, Transmit weight cells: 0

  Logical interface at-0/2/1.32767
    Flags: Point-To-Multipoint No-Multicast SNMP-Traps 0x4000
    Encapsulation: ATM-VCMUX
    VCI 0.4
      Flags: Active
      Total down time: 0 sec, Last down: Never
      EPD threshold: 0, Transmit weight cells: 0
```

show interfaces detail (ATM2, SDH Mode)

```
user@host> show interfaces at-0/2/1 detail
Physical interface: at-0/2/1, Enabled, Physical link is Up
  Interface index: 154, SNMP ifIndex: 42, Generation: 40
  Link-level type: ATM-PVC, MTU: 4482, Clocking: Internal, SDH mode, Speed: OC3,

  Loopback: None, Payload scrambler: Enabled
  Device flags   : Present Running
  Link flags     : None
  CoS queues     : 4 supported, 4 maximum usable queues
  Hold-times     : Up 0 ms, Down 0 ms
  Current address: 00:05:85:8f:30:3f
  Last flapped   : 2006-03-24 13:29:58 PST (00:05:10 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
  Egress queues: 4 supported, 4 in use
  Queue counters:       Queued packets  Transmitted packets      Dropped packets

    0 best-effort                    0                    0                    0

    1 expedited-fo                   0                    0                    0

    2 assured-forw                   0                    0                    0

    3 network-cont                   0                    0                    0

  SDH   alarms   : None
  SDH   defects  : None
    VPI 0
      Flags: Active
      Total down time: 0 sec, Last down: Never
        Traffic statistics:
         Input  bytes  :                    0
         Output bytes  :                    0
         Input  packets:                    0
         Output packets:                    0

  Logical interface at-0/2/1.0 (Index 75) (SNMP ifIndex 51) (Generation 25)
    Flags: Point-To-Point SNMP-Traps 0x4000 Encapsulation: ATM-SNAP
    Traffic statistics:
     Input  bytes  :                    0
     Output bytes  :                    0
     Input  packets:                    0
     Output packets:                    0
    Local statistics:
     Input  bytes  :                    0
     Output bytes  :                    0
     Input  packets:                    0
     Output packets:                    0
    Transit statistics:
     Input  bytes  :                    0                    0 bps
     Output bytes  :                    0                    0 bps
     Input  packets:                    0                    0 pps
     Output packets:                    0                    0 pps
    Protocol inet, MTU: 4470, Generation: 62, Route table: 0
      Flags: None
```

```
      Addresses, Flags: Is-Preferred Is-Primary
        Destination: 10.0.12.6, Local: 10.0.12.5, Broadcast: Unspecified,
        Generation: 58
    Protocol iso, MTU: 4470, Generation: 63, Route table: 0
      Flags: None
    VCI 0.128
      Flags: Active
      Total down time: 0 sec, Last down: Never
      EPD threshold: 2129, Transmit weight cells: 0
      ATM per-VC transmit statistics:
       Tail queue packet drops: 0
      Traffic statistics:
       Input  bytes  :                    0
       Output bytes  :                    0
       Input  packets:                    0
       Output packets:                    0
  Logical interface at-0/2/1.32767 (Index 76) (SNMP ifIndex 50) (Generation 26)
    Flags: Point-To-Multipoint No-Multicast SNMP-Traps 0x4000
    Encapsulation: ATM-VCMUX
    Traffic statistics:
     Input  bytes  :                    0
     Output bytes  :                    0
     Input  packets:                    0
     Output packets:                    0
    Local statistics:
     Input  bytes  :                    0
     Output bytes  :                    0
     Input  packets:                    0
     Output packets:                    0
    VCI 0.4
      Flags: Active
      Total down time: 0 sec, Last down: Never
      EPD threshold: 0, Transmit weight cells: 0
      ATM per-VC transmit statistics:
       Tail queue packet drops: 0
      Traffic statistics:
       Input  bytes  :                    0
       Output bytes  :                    0
       Input  packets:                    0
       Output packets:                    0
```

show interfaces extensive (ATM2, SDH Mode)

```
user@host> show interfaces at-0/2/1 extensive
Physical interface: at-0/2/1, Enabled, Physical link is Up
  Interface index: 154, SNMP ifIndex: 42, Generation: 40
  Link-level type: ATM-PVC, MTU: 4482, Clocking: Internal, SDH mode, Speed: OC3,

  Loopback: None, Payload scrambler: Enabled
  Device flags   : Present Running
  Link flags     : None
  CoS queues     : 4 supported, 4 maximum usable queues
  Hold-times     : Up 0 ms, Down 0 ms
  Current address: 00:05:85:8f:30:3f
  Last flapped   : 2006-03-24 13:29:58 PST (00:06:49 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 channel errors: 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
Egress queues: 4 supported, 4 in use
Queue counters:       Queued packets  Transmitted packets      Dropped packets

  0 best-effort                    0                    0                    0

  1 expedited-fo                   0                    0                    0

  2 assured-forw                   0                    0                    0

  3 network-cont                   0                    0                    0

SDH   alarms   : None
SDH   defects  : None
SDH PHY:              Seconds        Count  State
  PLL Lock                  0            0  OK
  PHY Light                 1            1  OK
SDH regenerator section:
  RS-BIP8                   2         8828
  OOF                       2            2  OK
  LOS                       2            1  OK
  LOF                       2            1  OK
  RS-ES                     4
  RS-SES                    3
  RS-SEFS                   2
SDH multiplex section:
  MS-BIP24                  2          771
  MS-FEBE                   1        17476
  MS-FERF                   2            1  OK
  MS-AIS                    2            1  OK
  BERR-SF                   0            0  OK
  BERR-SD                   0            0  OK
  MS-ES                     4
  MS-SES                    2
  MS-UAS                    0
  MS-ES-FE                  3
  MS-SES-FE                 2
  MS-UAS-FE                 0
SDH path:
  HP-BIP8                   1            6
  HP-FEBE                   1          251
  HP-LOP                    0            0  OK
  HP-AIS                    2            1  OK
  HP-FERF                   3            2  OK
  HP-UNEQ                   1            1  OK
  HP-PLM                    2            1  OK
  HP-ES                     4
  HP-SES                    3
  HP-UAS                    0
  HP-ES-FE                  3
  HP-SES-FE                 3
  HP-UAS-FE                 0
Received SDH overhead:
```



```
  F1      : 0x00, J0      : 0x00, K1      : 0x00, K2      : 0x00
  S1      : 0x00, C2      : 0x13, C2(cmp) : 0x13, F2      : 0x00
  Z3      : 0x00, Z4      : 0x00, S1(cmp) : 0x00
Transmitted SDH overhead:
  F1      : 0x00, J0      : 0x01, K1      : 0x00, K2      : 0x00
  S1      : 0x00, C2      : 0x13, F2      : 0x00, Z3      : 0x00
  Z4      : 0x00
ATM status:
  HCS state:     Sync
  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: 0, Output cell count: 0,
  Output idle cell count: 0, Output VC queue drops: 0, Input 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: 0
  VPI 0
    Flags: Active
    Total down time: 0 sec, Last down: Never
      Traffic statistics:
       Input  bytes  :                   0
       Output bytes  :                   0
       Input  packets:                   0
       Output packets:                   0

Logical interface at-0/2/1.0 (Index 75) (SNMP ifIndex 51) (Generation 25)
  Flags: Point-To-Point SNMP-Traps 0x4000 Encapsulation: ATM-SNAP
  Traffic statistics:
   Input  bytes  :                   0
   Output bytes  :                   0
   Input  packets:                   0
   Output packets:                   0
  Local statistics:
   Input  bytes  :                   0
   Output bytes  :                   0
   Input  packets:                   0
   Output packets:                   0
  Transit statistics:
   Input  bytes  :                   0                    0 bps
   Output bytes  :                   0                    0 bps
   Input  packets:                   0                    0 pps
   Output packets:                   0                    0 pps
  Protocol inet, MTU: 4470, Generation: 62, Route table: 0
    Flags: None
    Addresses, Flags: Is-Preferred Is-Primary
      Destination: 10.0.12.6, Local: 10.0.12.5, Broadcast: Unspecified,
      Generation: 58
  Protocol iso, MTU: 4470, Generation: 63, Route table: 0
    Flags: None
  VCI 0.128
    Flags: Active
    Total down time: 0 sec, Last down: Never
    EPD threshold: 2129, Transmit weight cells: 0
    ATM per-VC transmit statistics:
     Tail queue packet drops: 0
    Traffic statistics:
     Input  bytes  :                   0
     Output bytes  :                   0
```

```
      Input  packets:                    0
      Output packets:                    0
  Logical interface at-0/2/1.32767 (Index 76) (SNMP ifIndex 50) (Generation 26)
    Flags: Point-To-Multipoint No-Multicast SNMP-Traps 0x4000
    Encapsulation: ATM-VCMUX
    Traffic statistics:
     Input  bytes  :                   0
     Output bytes  :                   0
     Input  packets:                   0
     Output packets:                   0
    Local statistics:
     Input  bytes  :                   0
     Output bytes  :                   0
     Input  packets:                   0
     Output packets:                   0
    VCI 0.4
      Flags: Active
      Total down time: 0 sec, Last down: Never
      EPD threshold: 0, Transmit weight cells: 0
      ATM per-VC transmit statistics:
       Tail queue packet drops: 0
      Traffic statistics:
       Input  bytes  :                   0
       Output bytes  :                   0
       Input  packets:                   0
       Output packets:                   0
```

show interfaces (ATM2, SONET Mode)

```
user@host> show interfaces at-0/3/1
Physical interface: at-0/3/1, Enabled, Physical link is Up
  Interface index: 139, SNMP ifIndex: 67
  Link-level type: ATM-PVC, MTU: 4482, Clocking: Internal, SONET mode,
  Speed: OC3, Loopback: None, Payload scrambler: Enabled
  Device flags   : Present Running
  Link flags     : None
  CoS queues     : 4 supported, 4 maximum usable queues
  Current address: 00:14:f6:22:58:5e
  Last flapped   : 2006-03-13 17:46:36 PST (16:01:12 ago)
  Input rate     : 0 bps (0 pps)
  Output rate    : 0 bps (0 pps)
  SONET alarms   : None
  SONET defects  : None
    VPI 0
      Flags: Active, OAM, Shaping
      CBR, Peak: 50kbps
      OAM, Period 30 sec, Up count: 10, Down count: 10
      Total down time: 0 sec, Last down: Never
     OAM F4 cell statistics:
      Total received: 4, Total sent: 4
      Loopback received: 4, Loopback sent: 4
      RDI received: 0, RDI sent: 0
      AIS received: 0
  Traffic statistics:
         Input  packets:                    4
         Output packets:                   30
    VPI 10
      Flags: Active
      Total down time: 0 sec, Last down: Never
  Traffic statistics:
         Input  packets:                    0
```



```
       Output packets:                   0
  Logical interface at-0/3/1.0 (Index 78) (SNMP ifIndex 77)
    Flags: Point-To-Point Copy-PLP-To-CLP SNMP-Traps 0x4000
    Encapsulation: ATM-SNAP
    Input packets : 0
    Output packets: 0
    Protocol inet, MTU: 4470
      Flags: None
      Addresses, Flags: Is-Preferred Is-Primary
        Destination: 10.0.59.5, Local: 10.0.59.6
    Protocol iso, MTU: 4470
      Flags: None
    VCI 0.128
      Flags: Active
      Total down time: 0 sec, Last down: Never
      EPD threshold: 2129, Transmit weight cells: 10
        Input packets : 0
        Output packets: 0

  Logical interface at-0/3/1.32767 (Index 79) (SNMP ifIndex 76)
    Flags: Point-To-Multipoint Copy-PLP-To-CLP No-Multicast SNMP-Traps 0x4000
    Encapsulation: ATM-VCMUX
    Input packets : 4
    Output packets: 30
    VCI 0.16
      Flags: Active, ILMI
      Total down time: 0 sec, Last down: Never
      EPD threshold: 0, Transmit weight cells: 0
        Input packets : 0
        Output packets: 26
    VCI 0.4
      Flags: Active, OAM
      OAM, Period 30 sec, Up count: 10, Down count: 10
      Total down time: 0 sec, Last down: Never
      EPD threshold: 2129, Transmit weight cells: 0
        Input packets : 4
        Output packets: 4
     OAM F4 cell statistics:
      Total received: 4, Total sent: 4
      Loopback received: 4, Loopback sent: 4
      RDI received: 0, RDI sent: 0
      AIS received: 0, AIS sent: 0
```

show interfaces brief (ATM2, SONET Mode)

```
user@host> show interfaces at-0/3/1 brief
Physical interface: at-0/3/1, Enabled, Physical link is Up
  Link-level type: ATM-PVC, MTU: 4482, Clocking: Internal, SONET mode,
  Speed: OC3, Loopback: None, Payload scrambler: Enabled
  Device flags   : Present Running
  Link flags     : None

  Logical interface at-0/3/1.0
    Flags: Point-To-Point Copy-PLP-To-CLP SNMP-Traps 0x4000
    Encapsulation: ATM-SNAP
    inet  10.0.59.6        --> 10.0.59.5
    iso
    VCI 0.128
      Flags: Active
      Total down time: 0 sec, Last down: Never
      EPD threshold: 2129, Transmit weight cells: 10
```

```
  Logical interface at-0/3/1.32767
    Flags: Point-To-Multipoint Copy-PLP-To-CLP No-Multicast SNMP-Traps 0x4000
    Encapsulation: ATM-VCMUX
    VCI 0.16
      Flags: Active, ILMI
      Total down time: 0 sec, Last down: Never
      EPD threshold: 0, Transmit weight cells: 0
    VCI 0.4
      Flags: Active, OAM
      Total down time: 0 sec, Last down: Never
      EPD threshold: 2129, Transmit weight cells: 0
```

show interfaces detail (ATM2, SONET Mode)

```
user@host> show interfaces at-0/3/1 detail
Physical interface: at-0/3/1, Enabled, Physical link is Up
  Interface index: 139, SNMP ifIndex: 67, Generation: 22
  Link-level type: ATM-PVC, MTU: 4482, Clocking: Internal, SONET mode,
  Speed: OC3, Loopback: None, Payload scrambler: Enabled
  Device flags   : Present Running
  Link flags     : None
  CoS queues     : 4 supported, 4 maximum usable queues
  Hold-times     : Up 0 ms, Down 0 ms
  Current address: 00:14:f6:22:58:5e
  Last flapped   : 2006-03-13 17:46:36 PST (16:02:39 ago)
  Statistics last cleared: Never
  Traffic statistics:
   Input  bytes  :                  312                    0 bps
   Output bytes  :                 2952                    0 bps
   Input  packets:                    6                    0 pps
   Output packets:                   50                    0 pps
  Egress queues: 4 supported, 4 in use
  Queue counters:       Queued packets  Transmitted packets      Dropped packets

    0 best-effort                   44                   44                    0

    1 expedited-fo                   0                    0                    0

    2 assured-forw                   0                    0                    0

    3 network-cont                   6                    6                    0

  SONET alarms   : None
  SONET defects  : None
    VPI 0
      Flags: Active, OAM, Shaping
      CBR, Peak: 50kbps
      OAM, Period 30 sec, Up count: 10, Down count: 10
      Total down time: 0 sec, Last down: Never
     OAM F4 cell statistics:
      Total received: 6, Total sent: 6
      Loopback received: 6, Loopback sent: 6
      Last received: 00:00:29, Last sent: 00:00:29
      RDI received: 0, RDI sent: 0
      AIS received: 0
        Traffic statistics:
         Input  bytes  :                  312
         Output bytes  :                 2952
         Input  packets:                    6
         Output packets:                   50
```



```
    VPI 10
      Flags: Active
      Total down time: 0 sec, Last down: Never
        Traffic statistics:
         Input  bytes  :                    0
         Output bytes  :                    0
         Input  packets:                    0
         Output packets:                    0

  Logical interface at-0/3/1.0 (Index 78) (SNMP ifIndex 77) (Generation 20)
    Flags: Point-To-Point Copy-PLP-To-CLP SNMP-Traps 0x4000
    Encapsulation: ATM-SNAP
    Traffic statistics:
     Input  bytes  :                    0
     Output bytes  :                    0
     Input  packets:                    0
     Output packets:                    0
    Local statistics:
     Input  bytes  :                    0
     Output bytes  :                    0
     Input  packets:                    0
     Output packets:                    0
    Transit statistics:
     Input  bytes  :                    0                    0 bps
     Output bytes  :                    0                    0 bps
     Input  packets:                    0                    0 pps
     Output packets:                    0                    0 pps
    Protocol inet, MTU: 4470, Generation: 38, Route table: 0
      Flags: None
      Addresses, Flags: Is-Preferred Is-Primary
        Destination: 10.0.59.5, Local: 10.0.59.6, Broadcast: Unspecified,
        Generation: 44
    Protocol iso, MTU: 4470, Generation: 39, Route table: 0
      Flags: None
    VCI 0.128
      Flags: Active
      Total down time: 0 sec, Last down: Never
      EPD threshold: 2129, Transmit weight cells: 10
      ATM per-VC transmit statistics:
       Tail queue packet drops: 0
      Traffic statistics:
       Input  bytes  :                    0
       Output bytes  :                    0
       Input  packets:                    0
       Output packets:                    0
  Logical interface at-0/3/1.32767 (Index 79) (SNMP ifIndex 76) (Generation 21)
    Flags: Point-To-Multipoint Copy-PLP-To-CLP No-Multicast SNMP-Traps 0x4000
    Encapsulation: ATM-VCMUX
    Traffic statistics:
     Input  bytes  :                  360
     Output bytes  :                 3302
     Input  packets:                    6
     Output packets:                   50
    Local statistics:
     Input  bytes  :                  360
     Output bytes  :                 3302
     Input  packets:                    6
     Output packets:                   50
    VCI 0.16
      Flags: Active, ILMI
      Total down time: 0 sec, Last down: Never
```

```
      EPD threshold: 0, Transmit weight cells: 0
      ATM per-VC transmit statistics:
       Tail queue packet drops: 0
      Traffic statistics:
       Input  bytes  :                   0
       Output bytes  :                2640
       Input  packets:                   0
       Output packets:                  44
    VCI 0.4
      Flags: Active, OAM
      OAM, Period 30 sec, Up count: 10, Down count: 10
      Total down time: 0 sec, Last down: Never
      EPD threshold: 2129, Transmit weight cells: 0
      ATM per-VC transmit statistics:
       Tail queue packet drops: 0
      Traffic statistics:
       Input  bytes  :                 312
       Output bytes  :                 312
       Input  packets:                   6
       Output packets:                   6
     OAM F4 cell statistics:
      Total received: 6, Total sent: 6
      Loopback received: 6, Loopback sent: 6
      Last received: 00:00:29, Last sent: 00:00:29
      RDI received: 0, RDI sent: 0
      AIS received: 0, AIS sent: 0
```

show interfaces extensive (ATM2, SONET Mode)

```
user@host> show interfaces at-0/3/1 extensive
Physical interface: at-0/3/1, Enabled, Physical link is Up
  Interface index: 139, SNMP ifIndex: 67, Generation: 22
  Link-level type: ATM-PVC, MTU: 4482, Clocking: Internal, SONET mode,
  Speed: OC3, Loopback: None, Payload scrambler: Enabled
  Device flags   : Present Running
  Link flags     : None
  CoS queues     : 4 supported, 4 maximum usable queues
  Hold-times     : Up 0 ms, Down 0 ms
  Current address: 00:14:f6:22:58:5e
  Last flapped   : 2006-03-13 17:46:36 PST (16:04:12 ago)
  Statistics last cleared: Never
  Traffic statistics:
   Input  bytes  :                520                    0 bps
   Output bytes  :               4240                    0 bps
   Input  packets:                 10                    0 pps
   Output packets:                 72                    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: 1, Errors: 0, Drops: 0, Aged packets: 0, MTU errors: 0,

    Resource errors: 0
  Egress queues: 4 supported, 4 in use
  Queue counters:       Queued packets  Transmitted packets      Dropped packets

    0 best-effort                   62                   62                    0

    1 expedited-fo                   0                    0                    0
```



```
  2 assured-forw                 0                   0                   0

  3 network-cont                10                  10                   0

SONET alarms   : None
SONET defects  : None
SONET PHY:           Seconds        Count  State
  PLL Lock                 0            0  OK
  PHY Light                0            0  OK
SONET section:
  BIP-B1                   0            0
  SEF                      0            0  OK
  LOS                      0            0  OK
  LOF                      0            0  OK
  ES-S                     0
  SES-S                    0
  SEFS-S                   0
SONET line:
  BIP-B2                   0            0
  REI-L                    0            0
  RDI-L                    0            0  OK
  AIS-L                    0            0  OK
  BERR-SF                  0            0  OK
  BERR-SD                  0            0  OK
  ES-L                     0
  SES-L                    0
  UAS-L                    0
  ES-LFE                   0
  SES-LFE                  0
  UAS-LFE                  0
SONET path:
  BIP-B3                   0            0
  REI-P                    0            0
  LOP-P                    0            0  OK
  AIS-P                    0            0  OK
  RDI-P                    0            0  OK
  UNEQ-P                   1            1  OK
  PLM-P                    0            0  OK
  ES-P                     1
  SES-P                    1
  UAS-P                    0
  ES-PFE                   0
  SES-PFE                  0
  UAS-PFE                  0
Received SONET overhead:
  F1      : 0x00, J0      : 0x00, K1      : 0x00, K2      : 0x00
  S1      : 0x00, C2      : 0x13, C2(cmp) : 0x13, F2      : 0x00
  Z3      : 0x00, Z4      : 0x00, S1(cmp) : 0x00
Transmitted SONET overhead:
  F1      : 0x00, J0      : 0x01, K1      : 0x00, K2      : 0x00
  S1      : 0x00, C2      : 0x13, F2      : 0x00, Z3      : 0x00
  Z4      : 0x00
ATM status:
  HCS state:     Sync
  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: 0, Output cell count: 0,
  Output idle cell count: 0, Output VC queue drops: 0, Input 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: 0
  VPI 0
    Flags: Active, OAM, Shaping
    CBR, Peak: 50kbps
    OAM, Period 30 sec, Up count: 10, Down count: 10
    Total down time: 0 sec, Last down: Never
   OAM F4 cell statistics:
    Total received: 10, Total sent: 10
    Loopback received: 10, Loopback sent: 10
    Last received: 00:00:02, Last sent: 00:00:02
    RDI received: 0, RDI sent: 0
    AIS received: 0
      Traffic statistics:
       Input  bytes  :                 520
       Output bytes  :                4240
       Input  packets:                  10
       Output packets:                  72
  VPI 10
    Flags: Active
    Total down time: 0 sec, Last down: Never
      Traffic statistics:
       Input  bytes  :                   0
       Output bytes  :                   0
       Input  packets:                   0
       Output packets:                   0

Logical interface at-0/3/1.0 (Index 78) (SNMP ifIndex 77) (Generation 20)
  Flags: Point-To-Point Copy-PLP-To-CLP SNMP-Traps 0x4000
  Encapsulation: ATM-SNAP
  Traffic statistics:
   Input  bytes  :                   0
   Output bytes  :                   0
   Input  packets:                   0
   Output packets:                   0
  Local statistics:
   Input  bytes  :                   0
   Output bytes  :                   0
   Input  packets:                   0
   Output packets:                   0
  Transit statistics:
   Input  bytes  :                   0                    0 bps
   Output bytes  :                   0                    0 bps
   Input  packets:                   0                    0 pps
   Output packets:                   0                    0 pps
  Protocol inet, MTU: 4470, Generation: 38, Route table: 0
    Flags: None
    Addresses, Flags: Is-Preferred Is-Primary
      Destination: 10.0.59.5, Local: 10.0.59.6, Broadcast: Unspecified,
      Generation: 44
  Protocol iso, MTU: 4470, Generation: 39, Route table: 0
    Flags: None
  VCI 0.128
    Flags: Active
    Total down time: 0 sec, Last down: Never
    EPD threshold: 2129, Transmit weight cells: 10
    ATM per-VC transmit statistics:
     Tail queue packet drops: 0
    Traffic statistics:
```



```
     Input  bytes  :                    0
     Output bytes  :                    0
     Input  packets:                    0
     Output packets:                    0

Logical interface at-0/3/1.32767 (Index 79) (SNMP ifIndex 76) (Generation 21)
  Flags: Point-To-Multipoint Copy-PLP-To-CLP No-Multicast SNMP-Traps 0x4000
  Encapsulation: ATM-VCMUX
  Traffic statistics:
   Input  bytes  :                660
   Output bytes  :               5473
   Input  packets:                 11
   Output packets:                 83
  Local statistics:
   Input  bytes  :                660
   Output bytes  :               5473
   Input  packets:                 11
   Output packets:                 83
  VCI 0.16
    Flags: Active, ILMI
    Total down time: 0 sec, Last down: Never
    EPD threshold: 0, Transmit weight cells: 0
    ATM per-VC transmit statistics:
     Tail queue packet drops: 0
    Traffic statistics:
     Input  bytes  :                    0
     Output bytes  :                 4320
     Input  packets:                    0
     Output packets:                   72
  VCI 0.4
    Flags: Active, OAM
    OAM, Period 30 sec, Up count: 10, Down count: 10
    Total down time: 0 sec, Last down: Never
    EPD threshold: 2129, Transmit weight cells: 0
    ATM per-VC transmit statistics:
     Tail queue packet drops: 0
    Traffic statistics:
     Input  bytes  :                  572
     Output bytes  :                  572
     Input  packets:                   11
     Output packets:                   11
   OAM F4 cell statistics:
    Total received: 11, Total sent: 11
    Loopback received: 11, Loopback sent: 11
    Last received: 00:00:18, Last sent: 00:00:18
    RDI received: 0, RDI sent: 0
    AIS received: 0, AIS sent: 0
```

show interfaces (Gigabit Ethernet)

Syntax

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

Release Information

Command introduced before Junos OS Release 7.4.

Description

(M Series, T Series, and MX Series routers and EX Series switches only) Display status information about the specified Gigabit Ethernet interface.

Options

ge-*fpc/pic/port*—Display standard information about the specified Gigabit Ethernet interface.

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

descriptions—(Optional) Display interface description strings.

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

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

statistics—(Optional) Display static interface statistics.

Additional Information

In a logical system, this command displays information only about the logical interfaces and not about the physical interfaces.

Required Privilege Level

view

Related Documentation

- Verifying and Managing Agent Circuit Identifier-Based Dynamic VLAN Configuration on page 38

List of Sample Output

show interfaces (Gigabit Ethernet) on page 687
show interfaces (Gigabit Ethernet on MX Series Routers) on page 687
show interfaces extensive (Gigabit Ethernet on MX Series Routers showing interface transmit statistics configuration) on page 688
show interfaces brief (Gigabit Ethernet) on page 688
show interfaces detail (Gigabit Ethernet) on page 689
show interfaces extensive (Gigabit Ethernet IQ2) on page 690
show interfaces (Gigabit Ethernet Unnumbered Interface) on page 693
show interfaces (ACI Interface Set Configured) on page 693

Output Fields

Table 15 on page 673 describes the output fields for the **show interfaces** (Gigabit Ethernet) command. Output fields are listed in the approximate order in which they appear. For

Gigabit Ethernet IQ and IQE PICs, the traffic and MAC statistics vary by interface type. For more information, see Table 16 on page 686.

Table 15: show interfaces Gigabit Ethernet Output Fields

Field Name	Field Description	Level of Output
Physical Interface		
Physical interface	Name of the physical interface.	All levels
Enabled	State of the interface. Possible values are described in the “Enabled Field” section under *Common Output Fields Description*.	All levels
Interface index	Index number of the physical interface, which reflects its initialization sequence.	**detail extensive** none
SNMP ifIndex	SNMP index number for the physical interface.	**detail extensive** none
Generation	Unique number for use by Juniper Networks technical support only.	**detail extensive**
Link-level type	Encapsulation being used on the physical interface.	All levels
MTU	Maximum transmission unit size on the physical interface.	All levels
Speed	Speed at which the interface is running.	All levels
Loopback	Loopback status: **Enabled** or **Disabled**. If loopback is enabled, type of loopback: **Local** or **Remote**.	All levels
Source filtering	Source filtering status: **Enabled** or **Disabled**.	All levels
LAN-PHY mode	10-Gigabit Ethernet interface operating in Local Area Network Physical Layer Device (LAN PHY) mode. LAN PHY allows 10-Gigabit Ethernet wide area links to use existing Ethernet applications.	All levels
WAN-PHY mode	10-Gigabit Ethernet interface operating in Wide Area Network Physical Layer Device (WAN PHY) mode. WAN PHY allows 10-Gigabit Ethernet wide area links to use fiber-optic cables and other devices intended for SONET/SDH.	All levels
Unidirectional	Unidirectional link mode status for 10-Gigabit Ethernet interface: **Enabled** or **Disabled** for parent interface; **Rx-only** or **Tx-only** for child interfaces.	All levels
Flow control	Flow control status: **Enabled** or **Disabled**.	All levels
Auto-negotiation	(Gigabit Ethernet interfaces) Autonegotiation status: **Enabled** or **Disabled**.	All levels
Remote-fault	(Gigabit Ethernet interfaces) Remote fault status: • **Online**—Autonegotiation is manually configured as online. • **Offline**—Autonegotiation is manually configured as offline.	All levels
Device flags	Information about the physical device. Possible values are described in the “Device Flags” section under *Common Output Fields Description*.	All levels

Table 15: show interfaces Gigabit Ethernet Output Fields *(continued)*

Field Name	Field Description	Level of Output
Interface flags	Information about the interface. Possible values are described in the “Interface Flags” section under *Common Output Fields Description*.	All levels
Link flags	Information about the link. Possible values are described in the “Links Flags” section under *Common Output Fields Description*.	All levels
Wavelength	(10-Gigabit Ethernet dense wavelength-division multiplexing [DWDM] interfaces) Displays the configured wavelength, in nanometers (nm).	All levels
Frequency	(10-Gigabit Ethernet DWDM interfaces only) Displays the frequency associated with the configured wavelength, in terahertz (THz).	All levels
CoS queues	Number of CoS queues configured.	**detail extensive** none
Schedulers	(Gigabit Ethernet intelligent queuing 2 [IQ2] interfaces only) Number of CoS schedulers configured.	**extensive**
Hold-times	Current interface hold-time up and hold-time down, in milliseconds (ms).	**detail extensive**
Current address	Configured MAC address.	**detail extensive** none
Hardware address	Hardware MAC address.	**detail extensive** none
Last flapped	Date, time, and how long ago the interface went from down to up. The format is **Last flapped:** ***year-month-day hour:minute:second:timezone*** **(*hour:minute:second* ago)**. For example, **Last flapped: 2002-04-26 10:52:40 PDT (04:33:20 ago).**	**detail extensive** none
Input Rate	Input rate in bits per second (bps) and packets per second (pps). The value in this field also includes the Layer 2 overhead bytes for ingress traffic on Ethernet interfaces if you enable accounting of Layer 2 overhead at the PIC level or the logical interface level.	None
Output Rate	Output rate in bps and pps. The value in this field also includes the Layer 2 overhead bytes for egress traffic on Ethernet interfaces if you enable accounting of Layer 2 overhead at the PIC level or the logical interface level.	None
Statistics last cleared	Time when the statistics for the interface were last set to zero.	**detail extensive**
Egress account overhead	Layer 2 overhead in bytes that is accounted in the interface statistics for egress traffic.	**detail extensive**
Ingress account overhead	Layer 2 overhead in bytes that is accounted in the interface statistics for ingress traffic.	**detail extensive**

Table 15: show interfaces Gigabit Ethernet Output Fields *(continued)*

Field Name	Field Description	Level of Output
Traffic statistics	Number and rate of bytes and packets received and transmitted on the physical interface. • **Input bytes**—Number of bytes received on the interface. The value in this field also includes the Layer 2 overhead bytes for ingress traffic on Ethernet interfaces if you enable accounting of Layer 2 overhead at the PIC level or the logical interface level. • **Output bytes**—Number of bytes transmitted on the interface. The value in this field also includes the Layer 2 overhead bytes for egress traffic on Ethernet interfaces if you enable accounting of Layer 2 overhead at the PIC level or the logical interface level. • **Input packets**—Number of packets received on the interface. • **Output packets**—Number of packets transmitted on the interface. Gigabit Ethernet and 10-Gigabit Ethernet IQ PICs count the overhead and CRC bytes. For Gigabit Ethernet IQ PICs, the input byte counts vary by interface type. For more information, see Table 31 under the show interfaces (10-Gigabit Ethernet) command.	**detail extensive**
Input errors	Input errors on the interface. The following paragraphs explain the counters whose meaning might not be obvious: • **Errors**—Sum of the incoming frame aborts and FCS errors. • **Drops**—Number of packets dropped by the input queue of the I/O Manager ASIC. If the interface is saturated, this number increments once for every packet that is dropped by the ASIC's RED mechanism. • **Framing errors**—Number of packets received with an invalid frame checksum (FCS). • **Runts**—Number of frames received that are smaller than the runt threshold. • **Policed discards**—Number of frames that the incoming packet match code discarded because they were not recognized or not of interest. Usually, this field reports protocols that Junos OS does not handle. • **L3 incompletes**—Number of incoming packets discarded because they failed Layer 3 (usually IPv4) sanity checks of the header. For example, a frame with less than 20 bytes of available IP header is discarded. L3 incomplete errors can be ignored by configuring the **ignore-l3-incompletes** statement. • **L2 channel errors**—Number of times the software did not find a valid logical interface for an incoming frame. • **L2 mismatch timeouts**—Number of malformed or short packets that caused the incoming packet handler to discard the frame as unreadable. • **FIFO errors**—Number of FIFO errors in the receive direction that are reported by the ASIC on the PIC. If this value is ever nonzero, the PIC is probably malfunctioning. • **Resource errors**—Sum of transmit drops.	**extensive**

Table 15: show interfaces Gigabit Ethernet Output Fields *(continued)*

Field Name	Field Description	Level of Output
Output errors	Output errors on the interface. The following paragraphs explain the counters whose meaning might not be obvious: • **Carrier transitions**—Number of times the interface has gone from **down** to **up**. This number does not normally increment quickly, increasing only when the cable is unplugged, the far-end system is powered down and then up, or another problem occurs. If the number of carrier transitions increments quickly (perhaps once every 10 seconds), the cable, the far-end system, or the PIC or PIM is malfunctioning. • **Errors**—Sum of the outgoing frame aborts and FCS errors. • **Drops**—Number of packets dropped by the output queue of the I/O Manager ASIC. If the interface is saturated, this number increments once for every packet that is dropped by the ASIC's RED mechanism. NOTE: Due to accounting space limitations on certain Type 3 FPCs (which are supported in M320 and T640 routers), the **Drops** field does not always use the correct value for queue 6 or queue 7 for interfaces on 10-port 1-Gigabit Ethernet PICs. • **Collisions**—Number of Ethernet collisions. The Gigabit Ethernet PIC supports only full-duplex operation, so for Gigabit Ethernet PICs, this number should always remain 0. If it is nonzero, there is a software bug. • **Aged packets**—Number of packets that remained in shared packet SDRAM so long that the system automatically purged them. The value in this field should never increment. If it does, it is most likely a software bug or possibly malfunctioning hardware. • **FIFO errors**—Number of FIFO errors in the send direction as reported by the ASIC on the PIC. If this value is ever nonzero, the PIC is probably malfunctioning. • **HS link CRC errors**—Number of errors on the high-speed links between the ASICs responsible for handling the router interfaces. • **MTU errors**—Number of packets whose size exceeded the MTU of the interface. • **Resource errors**—Sum of transmit drops.	**extensive**
Egress queues	Total number of egress queues supported on the specified interface.	**detail extensive**
Queue counters (Egress)	CoS queue number and its associated user-configured forwarding class name. • **Queued packets**—Number of queued packets. • **Transmitted packets**—Number of transmitted packets. • **Dropped packets**—Number of packets dropped by the ASIC's RED mechanism. NOTE: Due to accounting space limitations on certain Type 3 FPCs (which are supported in M320 and T640 routers), the **Dropped packets** field does not always display the correct value for queue 6 or queue 7 for interfaces on 10-port 1-Gigabit Ethernet PICs.	**detail extensive**
Ingress queues	Total number of ingress queues supported on the specified interface. Displayed on IQ2 interfaces.	**extensive**

Table 15: show interfaces Gigabit Ethernet Output Fields *(continued)*

Field Name	Field Description	Level of Output
Queue counters (Ingress)	CoS queue number and its associated user-configured forwarding class name. Displayed on IQ2 interfaces. • **Queued packets**—Number of queued packets. • **Transmitted packets**—Number of transmitted packets. • **Dropped packets**—Number of packets dropped by the ASIC's RED mechanism.	**extensive**
Active alarms and Active defects	Ethernet-specific defects that can prevent the interface from passing packets. When a defect persists for a certain amount of time, it is promoted to an alarm. Based on the router configuration, an alarm can ring the red or yellow alarm bell on the router, or turn on the red or yellow alarm LED on the craft interface. These fields can contain the value **None** or **Link**. • **None**—There are no active defects or alarms. • **Link**—Interface has lost its link state, which usually means that the cable is unplugged, the far-end system has been turned off, or the PIC is malfunctioning.	**detail extensive** none
Interface transmit statistics	(On MX Series devices) Status of the **interface-transmit-statistics** configuration: Enabled or Disabled. • **Enabled**—When the **interface-transmit-statistics** statement is included in the configuration. If this is configured, the interface statistics show the actual transmitted load on the interface. • **Disabled**—When the **interface-transmit-statistics** statement is not included in the configuration. If this is not configured, the interface statistics show the offered load on the interface.	**detail extensive**
OTN FEC statistics	The forward error correction (FEC) counters provide the following statistics: • **Corrected Errors**—The count of corrected errors in the last second. • **Corrected Error Ratio**—The corrected error ratio in the last 25 seconds. For example, 1e-7 is 1 error per 10 million bits.	**detail extensive**
PCS statistics	(10-Gigabit Ethernet interfaces) Displays Physical Coding Sublayer (PCS) fault conditions from the WAN PHY or the LAN PHY device. • **Bit errors**—The number of seconds during which at least one bit error rate (BER) occurred while the PCS receiver is operating in normal mode. • **Errored blocks**—The number of seconds when at least one errored block occurred while the PCS receiver is operating in normal mode.	**detail extensive**

Table 15: show interfaces Gigabit Ethernet Output Fields *(continued)*

<table>
<tr><th>Field Name</th><th>Field Description</th><th>Level of Output</th></tr>
<tr><td>MAC statistics</td><td>Receive and Transmit statistics reported by the PIC's MAC subsystem, including the following:

Total octets and total packets—Total number of octets and packets. For Gigabit Ethernet IQ PICs, the received octets count varies by interface type. For more information, see Table 31 under the show interfaces (10-Gigabit Ethernet) command.
Unicast packets, Broadcast packets, and Multicast packets—Number of unicast, broadcast, and multicast packets.
CRC/Align errors—Total number of packets received that had a length (excluding framing bits, but including FCS octets) of between 64 and 1518 octets, inclusive, and had either a bad FCS with an integral number of octets (FCS Error) or a bad FCS with a nonintegral number of octets (Alignment Error).
FIFO error—Number of FIFO errors that are reported by the ASIC on the PIC. If this value is ever nonzero, the PIC or a cable is probably malfunctioning.
MAC control frames—Number of MAC control frames.
MAC pause frames—Number of MAC control frames with pause operational code.
Oversized frames—There are two possible conditions regarding the number of oversized frames:

Packet length exceeds 1518 octets, or
Packet length exceeds MRU

Jabber frames—Number of frames that were longer than 1518 octets (excluding framing bits, but including FCS octets), and had either an FCS error or an alignment error. This definition of jabber is different from the definition in IEEE-802.3 section 8.2.1.5 (10BASE5) and section 10.3.1.4 (10BASE2). These documents define jabber as the condition in which any packet exceeds 20 ms. The allowed range to detect jabber is from 20 ms to 150 ms.
Fragment frames—Total number of packets that were less than 64 octets in length (excluding framing bits, but including FCS octets) and had either an FCS error or an alignment error. Fragment frames normally increment because both runts (which are normal occurrences caused by collisions) and noise hits are counted.
VLAN tagged frames—Number of frames that are VLAN tagged. The system uses the TPID of 0x8100 in the frame to determine whether a frame is tagged or not.
NOTE: The 20-port Gigabit Ethernet MIC (MIC-3D-20GE-SFP) does not have hardware counters for VLAN frames. Therefore, the VLAN tagged frames field displays 0 when the show interfaces command is executed on a 20-port Gigabit Ethernet MIC. In other words, the number of VLAN tagged frames cannot be determined for the 20-port Gigabit Ethernet MIC.
Code violations—Number of times an event caused the PHY to indicate “Data reception error” or “invalid data symbol error.”
</td><td>extensive</td></tr>
<tr><td>OTN Received Overhead Bytes</td><td>APS/PCC0: 0x02, APS/PCC1: 0x11, APS/PCC2: 0x47, APS/PCC3: 0x58 Payload Type: 0x08</td><td>extensive</td></tr>
<tr><td>OTN Transmitted Overhead Bytes</td><td>APS/PCC0: 0x00, APS/PCC1: 0x00, APS/PCC2: 0x00, APS/PCC3: 0x00 Payload Type: 0x08</td><td>extensive</td></tr>
</table>

Table 15: show interfaces Gigabit Ethernet Output Fields *(continued)*

Field Name	Field Description	Level of Output
Filter statistics	**Receive** and **Transmit** statistics reported by the PIC's MAC address filter subsystem. The filtering is done by the content-addressable memory (CAM) on the PIC. The filter examines a packet's source and destination MAC addresses to determine whether the packet should enter the system or be rejected. • **Input packet count**—Number of packets received from the MAC hardware that the filter processed. • **Input packet rejects**—Number of packets that the filter rejected because of either the source MAC address or the destination MAC address. • **Input DA rejects**—Number of packets that the filter rejected because the destination MAC address of the packet is not on the accept list. It is normal for this value to increment. When it increments very quickly and no traffic is entering the router from the far-end system, either there is a bad ARP entry on the far-end system, or multicast routing is not on and the far-end system is sending many multicast packets to the local router (which the router is rejecting). • **Input SA rejects**—Number of packets that the filter rejected because the source MAC address of the packet is not on the accept list. The value in this field should increment only if source MAC address filtering has been enabled. If filtering is enabled, if the value increments quickly, and if the system is not receiving traffic that it should from the far-end system, it means that the user-configured source MAC addresses for this interface are incorrect. • **Output packet count**—Number of packets that the filter has given to the MAC hardware. • **Output packet pad count**—Number of packets the filter padded to the minimum Ethernet size (60 bytes) before giving the packet to the MAC hardware. Usually, padding is done only on small ARP packets, but some very small IP packets can also require padding. If this value increments rapidly, either the system is trying to find an ARP entry for a far-end system that does not exist or it is misconfigured. • **Output packet error count**—Number of packets with an indicated error that the filter was given to transmit. These packets are usually aged packets or are the result of a bandwidth problem on the FPC hardware. On a normal system, the value of this field should not increment. • **CAM destination filters, CAM source filters**—Number of entries in the CAM dedicated to destination and source MAC address filters. There can only be up to 64 source entries. If source filtering is disabled, which is the default, the values for these fields should be 0.	**extensive**
PMA PHY	(10-Gigabit Ethernet interfaces, WAN PHY mode) SONET error information: • **Seconds**—Number of seconds the defect has been active. • **Count**—Number of times that the defect has gone from inactive to active. • **State**—State of the error. Any state other than **OK** indicates a problem. Subfields are: • **PHY Lock**—Phase-locked loop • **PHY Light**—Loss of optical signal	**extensive**

Table 15: show interfaces Gigabit Ethernet Output Fields *(continued)*

Field Name	Field Description	Level of Output
WIS section	(10-Gigabit Ethernet interfaces, WAN PHY mode) SONET error information: • **Seconds**—Number of seconds the defect has been active. • **Count**—Number of times that the defect has gone from inactive to active. • **State**—State of the error. Any state other than **OK** indicates a problem. Subfields are: • **BIP-B1**—Bit interleaved parity for SONET section overhead • **SEF**—Severely errored framing • **LOL**—Loss of light • **LOF**—Loss of frame • **ES-S**—Errored seconds (section) • **SES-S**—Severely errored seconds (section) • **SEFS-S**—Severely errored framing seconds (section)	**extensive**
WIS line	(10-Gigabit Ethernet interfaces, WAN PHY mode) Active alarms and defects, plus counts of specific SONET errors with detailed information: • **Seconds**—Number of seconds the defect has been active. • **Count**—Number of times that the defect has gone from inactive to active. • **State**—State of the error. Any state other than **OK** indicates a problem. Subfields are: • **BIP-B2**—Bit interleaved parity for SONET line overhead • **REI-L**—Remote error indication (near-end line) • **RDI-L**—Remote defect indication (near-end line) • **AIS-L**—Alarm indication signal (near-end line) • **BERR-SF**—Bit error rate fault (signal failure) • **BERR-SD**—Bit error rate defect (signal degradation) • **ES-L**—Errored seconds (near-end line) • **SES-L**—Severely errored seconds (near-end line) • **UAS-L**—Unavailable seconds (near-end line) • **ES-LFE**—Errored seconds (far-end line) • **SES-LFE**—Severely errored seconds (far-end line) • **UAS-LFE**—Unavailable seconds (far-end line)	**extensive**

Table 15: show interfaces Gigabit Ethernet Output Fields *(continued)*

Field Name	Field Description	Level of Output
WIS path	(10-Gigabit Ethernet interfaces, WAN PHY mode) Active alarms and defects, plus counts of specific SONET errors with detailed information: • **Seconds**—Number of seconds the defect has been active. • **Count**—Number of times that the defect has gone from inactive to active. • **State**—State of the error. Any state other than **OK** indicates a problem. Subfields are: • **BIP-B3**—Bit interleaved parity for SONET section overhead • **REI-P**—Remote error indication • **LOP-P**—Loss of pointer (path) • **AIS-P**—Path alarm indication signal • **RDI-P**—Path remote defect indication • **UNEQ-P**—Path unequipped • **PLM-P**—Path payload (signal) label mismatch • **ES-P**—Errored seconds (near-end STS path) • **SES-P**—Severely errored seconds (near-end STS path) • **UAS-P**—Unavailable seconds (near-end STS path) • **SES-PFE**—Severely errored seconds (far-end STS path) • **UAS-PFE**—Unavailable seconds (far-end STS path)	**extensive**

Table 15: show interfaces Gigabit Ethernet Output Fields *(continued)*

Field Name	Field Description	Level of Output

Autonegotiation information	Information about link autonegotiation.
	• **Negotiation status:**
	  ◦ **Incomplete**—Ethernet interface has the speed or link mode configured.
	  ◦ **No autonegotiation**—Remote Ethernet interface has the speed or link mode configured, or does not perform autonegotiation.
	  ◦ **Complete**—Ethernet interface is connected to a device that performs autonegotiation and the autonegotiation process is successful.
	• **Link partner status**—**OK** when Ethernet interface is connected to a device that performs autonegotiation and the autonegotiation process is successful.
	• **Link partner**—Information from the remote Ethernet device:
	  ◦ **Link mode**—Depending on the capability of the link partner, either **Full-duplex** or **Half-duplex**.
	  ◦ **Flow control**—Types of flow control supported by the link partner. For Gigabit Ethernet interfaces, types are **Symmetric** (link partner supports **PAUSE** on receive and transmit), **Asymmetric** (link partner supports **PAUSE** on transmit), **Symmetric/Asymmetric** (link partner supports **PAUSE** on receive and transmit or only **PAUSE** on transmit), and **None** (link partner does not support flow control).
	  ◦ **Remote fault**—Remote fault information from the link partner—**Failure** indicates a receive link error. **OK** indicates that the link partner is receiving. **Negotiation error** indicates a negotiation error. **Offline** indicates that the link partner is going offline.
	• **Local resolution**—Information from the local Ethernet device:
	  ◦ **Flow control**—Types of flow control supported by the local device. For Gigabit Ethernet interfaces, advertised capabilities are **Symmetric/Asymmetric** (local device supports **PAUSE** on receive and transmit or only **PAUSE** on receive) and **None** (local device does not support flow control). Depending on the result of the negotiation with the link partner, local resolution flow control type will display **Symmetric** (local device supports **PAUSE** on receive and transmit), **Asymmetric** (local device supports **PAUSE** on receive), and **None** (local device does not support flow control).
	  ◦ **Remote fault**—Remote fault information. **Link OK** (no error detected on receive), **Offline** (local interface is offline), and **Link Failure** (link error detected on receive).
		extensive

Received path trace, Transmitted path trace	(10-Gigabit Ethernet interfaces, WAN PHY mode) SONET/SDH interfaces allow path trace bytes to be sent inband across the SONET/SDH link. Juniper Networks and other router manufacturers use these bytes to help diagnose misconfigurations and network errors by setting the transmitted path trace message so that it contains the system hostname and name of the physical interface. The received path trace value is the message received from the router at the other end of the fiber. The transmitted path trace value is the message that this router transmits.	**extensive**
Packet Forwarding Engine configuration	Information about the configuration of the Packet Forwarding Engine: • **Destination slot**—FPC slot number.	**extensive**

Table 15: show interfaces Gigabit Ethernet Output Fields *(continued)*

Field Name	Field Description	Level of Output
CoS information	Information about the CoS queue for the physical interface. • **CoS transmit queue**—Queue number and its associated user-configured forwarding class name. • **Bandwidth %**—Percentage of bandwidth allocated to the queue. • **Bandwidth bps**—Bandwidth allocated to the queue (in bps). • **Buffer %**—Percentage of buffer space allocated to the queue. • **Buffer usec**—Amount of buffer space allocated to the queue, in microseconds. This value is nonzero only if the buffer size is configured in terms of time. • **Priority**—Queue priority: **low** or **high**. • **Limit**—Displayed if rate limiting is configured for the queue. Possible values are **none** and **exact**. If **exact** is configured, the queue transmits only up to the configured bandwidth, even if excess bandwidth is available. If **none** is configured, the queue transmits beyond the configured bandwidth if bandwidth is available.	**extensive**
Logical Interface		
Logical interface	Name of the logical interface.	All levels
Index	Index number of the logical interface, which reflects its initialization sequence.	**detail extensive** none
SNMP ifIndex	SNMP interface index number for the logical interface.	**detail extensive** none
Generation	Unique number for use by Juniper Networks technical support only.	**detail extensive**
Flags	Information about the logical interface. Possible values are described in the “Logical Interface Flags” section under *Common Output Fields Description*.	All levels

Table 15: show interfaces Gigabit Ethernet Output Fields *(continued)*

Field Name	Field Description	Level of Output
VLAN-Tag	Rewrite profile applied to incoming or outgoing frames on the outer (**Out**) VLAN tag or for both the outer and inner (**In**) VLAN tags. • **push**—An outer VLAN tag is pushed in front of the existing VLAN tag. • **pop**—The outer VLAN tag of the incoming frame is removed. • **swap**—The outer VLAN tag of the incoming frame is overwritten with the user-specified VLAN tag information. • **push**—An outer VLAN tag is pushed in front of the existing VLAN tag. • **push-push**—Two VLAN tags are pushed in from the incoming frame. • **swap-push**—The outer VLAN tag of the incoming frame is replaced by a user-specified VLAN tag value. A user-specified outer VLAN tag is pushed in front. The outer tag becomes an inner tag in the final frame. • **swap-swap**—Both the inner and the outer VLAN tags of the incoming frame are replaced by the user-specified VLAN tag value. • **pop-swap**—The outer VLAN tag of the incoming frame is removed, and the inner VLAN tag of the incoming frame is replaced by the user-specified VLAN tag value. The inner tag becomes the outer tag in the final frame. • **pop-pop**—Both the outer and inner VLAN tags of the incoming frame are removed.	**brief detail extensive** none
Demux	IP demultiplexing (demux) value that appears if this interface is used as the demux underlying interface. The output is one of the following: • Source Family Inet • Destination Family Inet	**detail extensive** none
Encapsulation	Encapsulation on the logical interface.	All levels
ACI VLAN: Dynamic Profile	Name of the dynamic profile that defines the agent circuit identifier (ACI) interface set. If configured, the ACI interface set enables the underlying Ethernet interface to create dynamic VLAN subscriber interfaces based on ACI information.	**brief detail extensive** none
Protocol	Protocol family. Possible values are described in the “Protocol Field” section under *Common Output Fields Description*.	**detail extensive** none
MTU	Maximum transmission unit size on the logical interface.	**detail extensive** none
Dynamic Profile	(MX Series routers with Trio MPCs only) Name of the dynamic profile that was used to create this interface configured with a Point-to-Point Protocol over Ethernet (PPPoE) family.	**detail extensive** none
Service Name Table	(MX Series routers with Trio MPCs only) Name of the service name table for the interface configured with a PPPoE family.	**detail extensive** none
Max Sessions	(MX Series routers with Trio MPCs only) Maximum number of PPPoE logical interfaces that can be activated on the underlying interface.	**detail extensive** none

Table 15: show interfaces Gigabit Ethernet Output Fields *(continued)*

Field Name	Field Description	Level of Output
Duplicate Protection	(MX Series routers with Trio MPCs only) State of PPPoE duplicate protection: **On** or **Off**. When duplicate protection is configured for the underlying interface, a dynamic PPPoE logical interface cannot be activated when an existing active logical interface is present for the same PPPoE client.	**detail extensive** none
Direct Connect	State of the configuration to ignore DSL Forum VSAs: **On** or **Off**. When configured, the router ignores any of these VSAs received from a directly connected CPE device on the interface.	**detail extensive** none
AC Name	Name of the access concentrator.	**detail extensive** none
Maximum labels	Maximum number of MPLS labels configured for the MPLS protocol family on the logical interface.	**detail extensive** none
Traffic statistics	Number and rate of bytes and packets received and transmitted on the specified interface set. • **Input bytes, Output bytes**—Number of bytes received and transmitted on the interface set. The value in this field also includes the Layer 2 overhead bytes for ingress or egress traffic on Ethernet interfaces if you enable accounting of Layer 2 overhead at the PIC level or the logical interface level. • **Input packets, Output packets**—Number of packets received and transmitted on the interface set.	**detail extensive**
IPv6 transit statistics	Number of IPv6 transit bytes and packets received and transmitted on the logical interface if IPv6 statistics tracking is enabled.	**extensive**
Local statistics	Number and rate of bytes and packets destined to the router.	**extensive**
Transit statistics	Number and rate of bytes and packets transiting the switch. NOTE: For Gigabit Ethernet intelligent queuing 2 (IQ2) interfaces, the logical interface egress statistics might not accurately reflect the traffic on the wire when output shaping is applied. Traffic management output shaping might drop packets after they are tallied by the **Output bytes** and **Output packets** interface counters. However, correct values display for both of these egress statistics when per-unit scheduling is enabled for the Gigabit Ethernet IQ2 physical interface, or when a single logical interface is actively using a shared scheduler.	**extensive**
Generation	Unique number for use by Juniper Networks technical support only.	**detail extensive**
Route Table	Route table in which the logical interface address is located. For example, **0** refers to the routing table inet.0.	**detail extensive** none
Flags	Information about protocol family flags. Possible values are described in the “Family Flags” section under *Common Output Fields Description*.	**detail extensive**
Donor interface	(Unnumbered Ethernet) Interface from which an unnumbered Ethernet interface borrows an IPv4 address.	**detail extensive** none

Table 15: show interfaces Gigabit Ethernet Output Fields *(continued)*

Field Name	Field Description	Level of Output
Preferred source address	(Unnumbered Ethernet) Secondary IPv4 address of the donor loopback interface that acts as the preferred source address for the unnumbered Ethernet interface.	**detail extensive** none
Input Filters	Names of any input filters applied to this interface. If you specify a precedence value for any filter in a dynamic profile, filter precedence values appear in parentheses next to all interfaces.	**detail extensive**
Output Filters	Names of any output filters applied to this interface. If you specify a precedence value for any filter in a dynamic profile, filter precedence values appear in parentheses next to all interfaces.	**detail extensive**
Mac-Validate Failures	Number of MAC address validation failures for packets and bytes. This field is displayed when MAC address validation is enabled for the logical interface.	**detail extensive** none
Addresses, Flags	Information about the address flags. Possible values are described in the “Addresses Flags” section under *Common Output Fields Description*.	**detail extensive** none
protocol-family	Protocol family configured on the logical interface. If the protocol is **inet**, the IP address of the interface is also displayed.	**brief**
Flags	Information about the address flag. Possible values are described in the “Addresses Flags” section under *Common Output Fields Description*.	**detail extensive** none
Destination	IP address of the remote side of the connection.	**detail extensive** none
Local	IP address of the logical interface.	**detail extensive** none
Broadcast	Broadcast address of the logical interface.	**detail extensive** none
Generation	Unique number for use by Juniper Networks technical support only.	**detail extensive**

Table 16: Gigabit Ethernet IQ PIC Traffic and MAC Statistics by Interface Type

Interface Type	Sample Command	Byte and Octet Counts Include	Comments
Inbound physical interface	**show interfaces ge-0/3/0 extensive**	Traffic statistics: Input bytes: 496 bytes per packet, representing the Layer 2 packet MAC statistics: Received octets: 500 bytes per packet, representing the Layer 2 packet + 4 bytes	The additional 4 bytes are for the CRC.
Inbound logical interface	**show interfaces ge-0/3/0.50 extensive**	Traffic statistics: Input bytes: 478 bytes per packet, representing the Layer 3 packet	

Table 16: Gigabit Ethernet IQ PIC Traffic and MAC Statistics by Interface Type *(continued)*

Interface Type	Sample Command	Byte and Octet Counts Include	Comments
Outbound physical interface	**show interfaces ge-0/0/0 extensive**	Traffic statistics: Input bytes: 490 bytes per packet, representing the Layer 3 packet + 12 bytes MAC statistics: Received octets: 478 bytes per packet, representing the Layer 3 packet	For input bytes, the additional 12 bytes include 6 bytes for the destination MAC address plus 4 bytes for VLAN plus 2 bytes for the Ethernet type.
Outbound logical interface	**show interfaces ge-0/0/0.50 extensive**	Traffic statistics: Input bytes: 478 bytes per packet, representing the Layer 3 packet	

Sample Output

show interfaces (Gigabit Ethernet)

```
user@host> show interfaces ge-3/0/2
Physical interface: ge-3/0/2, Enabled, Physical link is Up
  Interface index: 167, SNMP ifIndex: 35
  Link-level type: 52, MTU: 1522, Speed: 1000mbps, Loopback: Disabled,
  Source filtering: Disabled, Flow control: Enabled, Auto-negotiation: Enabled
  Remote fault: Online
  Device flags   : Present Running
  Interface flags: SNMP-Traps Internal: 0x4000
  CoS queues     : 4 supported, 4 maximum usable queues
  Current address: 00:05:85:4a:e9:7c, Hardware address: 00:05:85:4a:e9:7c
  Last flapped   : 2006-08-10 17:25:10 PDT (00:01:08 ago)
  Input rate     : 0 bps (0 pps)
  Output rate    : 0 bps (0 pps)
  Ingress rate at Packet Forwarding Engine      : 0 bps (0 pps)
  Ingress drop rate at Packet Forwarding Engine : 0 bps (0 pps)
  Active alarms  : None
  Active defects : None

  Logical interface ge-3/0/2.0 (Index 72) (SNMP ifIndex 69)
    Flags: SNMP-Traps 0x4000
    VLAN-Tag [ 0x8100.512 0x8100.513 ] In(pop-swap 0x8100.530) Out(swap-push
    0x8100.512 0x8100.513)
    Encapsulation: VLAN-CCC
    Egress account overhead: 100
    Ingress account overhead: 90
    Input packets : 0
    Output packets: 0
    Protocol ccc, MTU: 1522
      Flags: Is-Primary
```

show interfaces (Gigabit Ethernet on MX Series Routers)

```
user@host> show interfaces ge-2/2/2
Physical interface: ge-2/2/2, Enabled, Physical link is Up
  Interface index: 156, SNMP ifIndex: 188
  Link-level type: Ethernet, MTU: 1514, Speed: 1000mbps, 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: 0x4000
  Link flags     : None
  CoS queues     : 8 supported, 4 maximum usable queues
  Schedulers     : 0
  Current address: 00:1f:12:b7:d7:c0, Hardware address: 00:1f:12:b7:d6:76
  Last flapped   : 2008-09-05 16:44:30 PDT (3d 01:04 ago)
  Input rate     : 0 bps (0 pps)
  Output rate    : 0 bps (0 pps)
  Active alarms  : None
  Active defects : None
  Logical interface ge-2/2/2.0 (Index 82) (SNMP ifIndex 219)
    Flags: SNMP-Traps 0x20000000 Encapsulation: Ethernet-Bridge
    Egress account overhead: 100
    Ingress account overhead: 90
    Input packets : 0
    Output packets: 0
     Protocol aenet, AE bundle: ae0.0    Link Index: 4
```

show interfaces extensive (Gigabit Ethernet on MX Series Routers showing interface transmit statistics configuration)

```
user@host> show interfaces ge-2/1/2 extensive | match "output|interface"
Physical interface: ge-2/1/2, Enabled, Physical link is Up
  Interface index: 151, SNMP ifIndex: 530, Generation: 154
  Interface flags: SNMP-Traps Internal: 0x4000
   Output bytes  :         240614363944            772721536 bps
   Output packets:           3538446506              1420444 pps
    Direction : Output
  Interface transmit statistics: Enabled

  Logical interface ge-2/1/2.0 (Index 331) (SNMP ifIndex 955) (Generation 146)
     Output bytes  :         195560312716            522726272 bps
     Output packets:           4251311146              1420451 pps
```

show interfaces brief (Gigabit Ethernet)

```
user@host> show interfaces ge-3/0/2 brief
Physical interface: ge-3/0/2, Enabled, Physical link is Up
  Link-level type: 52, MTU: 1522, Speed: 1000mbps, Loopback: Disabled,
  Source filtering: Disabled, Flow control: Enabled, Auto-negotiation: Enabled,
  Remote fault: Online
  Device flags   : Present Running
  Interface flags: SNMP-Traps Internal: 0x4000
  Link flags     : None

  Logical interface ge-3/0/2.0
    Flags: SNMP-Traps 0x4000
    VLAN-Tag [ 0x8100.512 0x8100.513 ] In(pop-swap 0x8100.530) Out(swap-push
    0x8100.512 0x8100.513)
    Encapsulation: VLAN-CCC
    ccc

  Logical interface ge-3/0/2.32767
    Flags: SNMP-Traps 0x4000 VLAN-Tag [ 0x0000.0 ]  Encapsulation: ENET2
```

show interfaces detail (Gigabit Ethernet)

user@host> **show interfaces ge-3/0/2 detail**

```
Physical interface: ge-3/0/2, Enabled, Physical link is Up
  Interface index: 167, SNMP ifIndex: 35, Generation: 177
  Link-level type: 52, MTU: 1522, Speed: 1000mbps, Loopback: Disabled,
  Source filtering: Disabled, Flow control: Enabled, Auto-negotiation: Enabled,
  Remote fault: Online
  Device flags   : Present Running
  Interface flags: SNMP-Traps Internal: 0x4000
  Link flags     : None
  CoS queues     : 4 supported, 4 maximum usable queues
  Hold-times     : Up 0 ms, Down 0 ms
  Current address: 00:05:85:4a:e9:7c, Hardware address: 00:05:85:4a:e9:7c
  Last flapped   : 2006-08-09 17:17:00 PDT (01:31:33 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
  Ingress traffic statistics at Packet Forwarding Engine:
   Input  bytes  :                    0                    0 bps
   Input  packets:                    0                    0 pps
   Drop   bytes  :                    0                    0 bps
   Drop   packets:                    0                    0 pps
  Ingress queues: 4 supported, 4 in use
  Queue counters:       Queued packets  Transmitted packets      Dropped packets

    0 best-effort                  0                    0                    0

    1 expedited-fo                 0                    0                    0

    2 assured-forw                 0                    0                    0

    3 network-cont                 0                    0                    0

  Egress queues: 4 supported, 4 in use
  Queue counters:       Queued packets  Transmitted packets      Dropped packets

    0 best-effort                  0                    0                    0

    1 expedited-fo                 0                    0                    0

    2 assured-forw                 0                    0                    0

    3 network-cont                 0                    0                    0

  Active alarms  : None
  Active defects : None

  Logical interface ge-3/0/2.0 (Index 72) (SNMP ifIndex 69) (Generation 140)
    Flags: SNMP-Traps 0x4000
    VLAN-Tag [0x8100.512 0x8100.513 ] In(pop-swap 0x8100.530)
Out(swap-push 0x8100.512 0x8100.513)
    Encapsulation: VLAN-CCC
    Egress account overhead: 100
    Ingress account overhead: 90
    Traffic statistics:
     Input  bytes  :                    0
     Output bytes  :                    0
```

```
      Input  packets:                 0
      Output packets:                 0
     Local statistics:
      Input  bytes  :                 0
      Output bytes  :                 0
      Input  packets:                 0
      Output packets:                 0
     Transit statistics:
      Input  bytes  :                 0                    0 bps
      Output bytes  :                 0                    0 bps
      Input  packets:                 0                    0 pps
      Output packets:                 0                    0 pps
     Protocol ccc, MTU: 1522, Generation: 149, Route table: 0
       Flags: Is-Primary

   Logical interface ge-3/0/2.32767 (Index 71) (SNMP ifIndex 70)
   (Generation 139)
     Flags: SNMP-Traps 0x4000 VLAN-Tag [ 0x0000.0 ]  Encapsulation: ENET2
     Traffic statistics:
      Input  bytes  :                 0
      Output bytes  :                 0
      Input  packets:                 0
      Output packets:                 0
     Local statistics:
      Input  bytes  :                 0
      Output bytes  :                 0
      Input  packets:                 0
      Output packets:                 0
     Transit statistics:
      Input  bytes  :                 0                    0 bps
      Output bytes  :                 0                    0 bps
      Input  packets:                 0                    0 pps
      Output packets:                 0                    0 pps
```

show interfaces extensive (Gigabit Ethernet IQ2)

```
user@host> show interfaces ge-7/1/3 extensive
Physical interface: ge-7/1/3, Enabled, Physical link is Up
  Interface index: 170, SNMP ifIndex: 70, Generation: 171
  Link-level type: Ethernet, MTU: 1514, Speed: 1000mbps, Loopback: Disabled,
  Source filtering: Disabled, Flow control: Enabled, Auto-negotiation: Enabled,
  Remote fault: Online
  Device flags   : Present Running
  Interface flags: SNMP-Traps Internal: 0x4004000
  Link flags     : None
  CoS queues     : 8 supported, 4 maximum usable queues
  Schedulers     : 256
  Hold-times     : Up 0 ms, Down 0 ms
  Current address: 00:14:f6:30:5e:74, Hardware address: 00:14:f6:30:5e:74
  Last flapped   : 2007-11-07 21:31:41 PST (02:03:33 ago)
  Statistics last cleared: Never
  Traffic statistics:
   Input  bytes  :          38910844056                 7952 bps
   Output bytes  :              7174605                 8464 bps
   Input  packets:            418398473                   11 pps
   Output packets:                78903                   12 pps
   IPv6 transit statistics:
    Input  bytes  :                   0
    Output bytes  :                   0
    Input  packets:                   0
    Output packets:                   0
```

```
Ingress traffic statistics at Packet Forwarding Engine:
 Input  bytes  :         38910799145                7952 bps
 Input  packets:           418397956                  11 pps
 Drop   bytes  :                   0                   0 bps
 Drop   packets:                   0                   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
Ingress queues: 4 supported, 4 in use
Queue counters:       Queued packets  Transmitted packets      Dropped packets

  0 best-effort           418390823            418390823                    0

  1 expedited-fo                  0                    0                    0

  2 assured-forw                  0                    0                    0

  3 network-cont               7133                 7133                    0

Egress queues: 4 supported, 4 in use
Queue counters:       Queued packets  Transmitted packets      Dropped packets

  0 best-effort                1031                 1031                    0

  1 expedited-fo                  0                    0                    0

  2 assured-forw                  0                    0                    0

  3 network-cont              77872                77872                    0

Active alarms  : None
Active defects : None
MAC statistics:                      Receive         Transmit
  Total octets                   38910844056          7174605
  Total packets                    418398473            78903
  Unicast packets               408021893366             1026
  Broadcast packets                       10               12
  Multicast packets                418398217            77865
  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  OTN Received Overhead Bytes:
  APS/PCC0: 0x02, APS/PCC1: 0x11, APS/PCC2: 0x47, APS/PCC3: 0x58
  Payload Type: 0x08
OTN Transmitted Overhead Bytes:
  APS/PCC0: 0x00, APS/PCC1: 0x00, APS/PCC2: 0x00, APS/PCC3: 0x00
  Payload Type: 0x08
Filter statistics:
  Input packet count               418398473
  Input packet rejects                   479
  Input DA rejects                       479
```

```
    Input SA rejects                        0
    Output packet count                                 78903
    Output packet pad count                                 0
    Output packet error count                               0
    CAM destination filters: 0, CAM source filters: 0
  Autonegotiation information:
    Negotiation status: Complete
    Link partner:
        Link mode: Full-duplex, Flow control: Symmetric/Asymmetric,
        Remote fault: OK
    Local resolution:
        Flow control: Symmetric, Remote fault: Link OK
  Packet Forwarding Engine configuration:
    Destination slot: 7
  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
    Direction : Input
    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

  Logical interface ge-7/1/3.0 (Index 70) (SNMP ifIndex 85) (Generation 150)
    Flags: SNMP-Traps Encapsulation: ENET2
    Traffic statistics:
     Input  bytes  :              812400
     Output bytes  :             1349206
     Input  packets:                9429
     Output packets:                9449
     IPv6 transit statistics:
      Input  bytes  :                  0
      Output bytes  :                  0
      Input  packets:                  0
      Output packets:                  0
    Local statistics:
     Input  bytes  :              812400
     Output bytes  :             1349206
     Input  packets:                9429
     Output packets:                9449
    Transit statistics:
     Input  bytes  :                   0                 7440 bps
     Output bytes  :                   0                 7888 bps
     Input  packets:                   0                   10 pps
     Output packets:                   0                   11 pps
     IPv6 transit statistics:
      Input  bytes  :                  0
      Output bytes  :                  0
      Input  packets:                  0
      Output packets:                  0
    Protocol inet, MTU: 1500, Generation: 169, Route table: 0
      Flags: Is-Primary, Mac-Validate-Strict
      Mac-Validate Failures: Packets: 0, Bytes: 0
      Addresses, Flags: Is-Preferred Is-Primary
```



```
    Input Filters: F1-ge-3/0/1.0-in, F3-ge-3/0/1.0-in
    Output Filters: F2-ge-3/0/1.0-out (53)
    Destination: 10.74.2/24, Local: 10.74.2.2, Broadcast: 10.74.2.255,
      Generation: 196
  Protocol multiservice, MTU: Unlimited, Generation: 170, Route table: 0
    Flags: Is-Primary
    Policer: Input: __default_arp_policer__
```

NOTE: For Gigabit Ethernet intelligent queuing 2 (IQ2) interfaces, the logical interface egress statistics displayed in the **show interfaces** command output might not accurately reflect the traffic on the wire when output shaping is applied. Traffic management output shaping might drop packets after they are tallied by the interface counters. For detailed information, see the description of the logical interface **Transit statistics** fields in Table 15 on page 673.

show interfaces (Gigabit Ethernet Unnumbered Interface)

```
user@host> show interfaces ge-3/2/0
Physical interface: ge-3/2/0, Enabled, Physical link is Up
  Interface index: 148, SNMP ifIndex: 50
  Link-level type: Ethernet, MTU: 1514, Speed: 1000mbps, Loopback: Disabled,
  Source filtering: Disabled, Flow control: Enabled, Auto-negotiation: Enabled,
  Remote fault: Online
  Device flags   : Present Running
  Interface flags: SNMP-Traps Internal: 0x4000
  Link flags     : None
  CoS queues     : 8 supported, 4 maximum usable queues
  Current address: 00:14:f6:11:26:f8, Hardware address: 00:14:f6:11:26:f8
  Last flapped   : 2006-10-27 04:42:23 PDT (08:01:52 ago)
  Input rate     : 0 bps (0 pps)
  Output rate    : 624 bps (1 pps)
  Active alarms  : None
  Active defects : None

  Logical interface ge-3/2/0.0 (Index 67) (SNMP ifIndex 85)
    Flags: SNMP-Traps Encapsulation: ENET2
    Input packets : 0
    Output packets: 6
    Protocol inet, MTU: 1500
      Flags: Unnumbered
      Donor interface: lo0.0 (Index 64)
      Preferred source address: 22.22.22.22
```

show interfaces (ACI Interface Set Configured)

```
user@host> show interfaces ge-1/0/0.4001
  Logical interface ge-1/0/0.4001 (Index 340) (SNMP ifIndex 548)
    Flags: SNMP-Traps 0x4000 VLAN-Tag [ 0x8100.4001 ]  Encapsulation: PPP-over-

    Ethernet
    ACI VLAN:
      Dynamic Profile: aci-vlan-set-profile
    PPPoE:
      Dynamic Profile: aci-vlan-pppoe-profile,
      Service Name Table: None,
      Max Sessions: 32000, Max Sessions VSA Ignore: Off,
      Duplicate Protection: On, Short Cycle Protection: Off,
      Direct Connect: Off,
      AC Name: nbc
```



```
Input packets : 9
Output packets: 8
Protocol multiservice, MTU: Unlimited
```


show interfaces (PPPoE)

Syntax

show interfaces pp0.*logical*
<brief | detail | extensive | terse>
<descriptions>
<media>
<snmp-index *snmp-index*>
<statistics>

Release Information

Command introduced before Junos OS Release 7.4.

Description

(J Series Services Routers, M120 routers, M320 routers, and MX Series routers only) Display status information about the PPPoE interface.

Options

pp0.*logical*—Display standard status information about the PPPoE interface.

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

descriptions—(Optional) Display interface description strings.

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

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

statistics—(Optional) Display PPPoE interface statistics.

Required Privilege Level

view

List of Sample Output

show interfaces (PPPoE) on page 701
show interfaces (PPPoE over Aggregated Ethernet) on page 701
show interfaces brief (PPPoE) on page 702
show interfaces detail (PPPoE) on page 702
show interfaces detail (PPPoE on J Series Services Routers) on page 703
show interfaces extensive (PPPoE on M120 and M320 Routers) on page 704

Output Fields

Table 17 on page 695 lists the output fields for the **show interfaces** (PPPoE) command. Output fields are listed in the approximate order in which they appear.

Table 17: show interfaces (PPPoE) Output Fields

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

Table 17: show interfaces (PPPoE) Output Fields *(continued)*

Field Name	Field Description	Level of Output
SNMP ifIndex	SNMP index number for the physical interface.	**detail extensive** none
Generation	Unique number for use by Juniper Networks technical support only.	**detail extensive**
Type	Physical interface type (PPPoE).	All levels
Link-level type	Encapsulation on the physical interface (PPPoE).	All levels
MTU	MTU size on the physical interface.	All levels
Clocking	Reference clock source. It can be **Internal** or **External**.	All levels
Speed	Speed at which the interface is running.	All levels
Device flags	Information about the physical device. Possible values are described in the “Device Flags” section under *Common Output Fields Description*.	All levels
Interface flags	Information about the interface. Possible values are described in the “Interface Flags” section under *Common Output Fields Description*.	All levels
Link type	Physical interface link type: **full duplex** or **half duplex**.	All levels
Link flags	Information about the interface. Possible values are described in the “Link Flags” section under *Common Output Fields Description*.	All levels
Input rate	Input rate in bits per second (bps) and packets per second (pps).	None specified
Output rate	Output rate in bps and pps.	None specified
Physical Info	Physical interface information.	All levels
Hold-times	Current interface hold-time up and hold-time down, in milliseconds.	**detail extensive**
Current address	Configured MAC address.	**detail extensive**
Hardware address	MAC address of the hardware.	**detail extensive**
Alternate link address	Backup address of the link.	**detail extensive**
Statistics last cleared	Time when the statistics for the interface were last set to zero.	**detail extensive**

Table 17: show interfaces (PPPoE) Output Fields *(continued)*

Field Name	Field Description	Level of Output
Traffic statistics	Number and rate of bytes and packets received and transmitted on the physical interface. • **Input bytes**—Number of bytes received on the interface. • **Output bytes**—Number of bytes transmitted on the interface. • **Input packets**—Number of packets received on the interface. • **Output packets**—Number of packets transmitted on the interface.	**detail extensive**
IPv6 transit statistics	Number of IPv6 transit bytes and packets received and transmitted on the physical interface if IPv6 statistics tracking is enabled. NOTE: These fields include dropped traffic and exception traffic, as those fields are not separately defined. • **Input bytes**—Number of bytes received on the interface. • **Output bytes**—Number of bytes transmitted on the interface. • **Input packets**—Number of packets received on the interface. • **Output packets**—Number of packets transmitted on the interface.	**detail extensive**
Input errors	Input errors on the interface: • **Errors**—Sum of incoming frame aborts and FCS errors. • **Drops**—Number of packets dropped by the input queue of the I/O Manager ASIC. If the interface is saturated, this number increments once for every packet that is dropped by the ASIC's RED mechanism. • **Framing errors**—Number of packets received with an invalid frame checksum (FCS). • **Runts**—Number of frames received that are smaller than the runt threshold. • **Giants**—Number of frames received that are larger than the giant threshold. • **Policed discards**—Number of frames that the incoming packet match code discarded because they were not recognized or not of interest. Usually, this field reports protocols that the Junos OS does not handle. • **Resource errors**—Sum of B chip Tx drops and IXP Tx net transmit drops.	**extensive**
Output errors	Output errors on the interface. The following paragraphs explain the counters whose meaning might not be obvious: • **Carrier transitions** —Number of times the interface has gone from **down** to **up**. This number does not normally increment quickly, increasing only when the cable is unplugged, the far-end system is powered down and then up, or another problem occurs. If the number of carrier transitions increments quickly (perhaps once every 10 seconds), then the cable, the far-end system, or the PIM is malfunctioning. • **Errors**—Sum of the outgoing frame aborts and FCS errors. • **Drops**—Number of packets dropped by the output queue of the I/O Manager ASIC. If the interface is saturated, this number increments once for every packet that is dropped by the ASIC's RED mechanism. • **MTU errors**—Number of packets whose size exceeded the MTU of the interface. • **Resource errors**—Sum of B chip Tx drops and IXP Tx net transmit drops.	**extensive**
Logical Interface		

Table 17: show interfaces (PPPoE) Output Fields *(continued)*

Field Name	Field Description	Level of Output
Logical interface	Name of the logical interface.	All levels
Index	Logical interface index number (which reflects its initialization sequence).	**detail extensive** none
SNMP ifIndex	Logical interface SNMP interface index number.	**detail extensive** none
Generation	Unique number for use by Juniper Networks technical support only.	**detail extensive**
Flags	Information about the logical interface. Possible values are described in the "Logical Interface Flags" section under *Common Output Fields Description*.	All levels
Encapsulation	Type of encapsulation configured on the logical interface.	All levels
PPP parameters	PPP status: • LCP restart timer—Length of time (in milliseconds) between successive Link Control Protocol (LCP) configuration requests. • NCP restart timer—Length of time (in milliseconds) between successive Network Control Protocol (NCP) configuration requests.	**detail**
PPPoE	PPPoE status: • **State**—State of the logical interface (**up** or **down**). • **Session ID**—PPPoE session ID. • **Service name**—Type of service required. Can be used to indicate an Internet service provider (ISP) name or a class or quality of service. • **Configured AC name**—Configured access concentrator name. • **Auto-reconnect timeout**—Time after which to try to reconnect after a PPPoE session is terminated, in seconds. • **Idle Timeout**—Length of time (in seconds) that a connection can be idle before disconnecting. • **Underlying interface**—Interface on which PPPoE is running.	All levels
Link	Name of the physical interfaces for member links in an aggregated Ethernet bundle for a PPPoE over aggregated Ethernet configuration. PPPoE traffic goes out on these interfaces.	All levels
Traffic statistics	Total number of bytes and packets received and transmitted on the logical interface. These statistics are the sum of the local and transit statistics. When a burst of traffic is received, the value in the output packet rate field might briefly exceed the peak cell rate. This counter usually takes less than 1 second to stabilize.	**detail extensive**

Table 17: show interfaces (PPPoE) Output Fields *(continued)*

Field Name	Field Description	Level of Output
IPv6 transit statistics	Number of IPv6 transit bytes and packets received and transmitted on the logical interface if IPv6 statistics tracking is enabled. NOTE: The packet and byte counts in these fields include traffic that is dropped and does not leave the router. • **Input bytes**—Number of bytes received on the interface. • **Output bytes**—Number of bytes transmitted on the interface. • **Input packets**—Number of packets received on the interface. • **Output packets**—Number of packets transmitted on the interface.	**detail extensive**
Local statistics	Statistics for traffic received from and transmitted to the Routing Engine. When a burst of traffic is received, the value in the output packet rate field might briefly exceed the peak cell rate. This counter usually takes less than 1 second to stabilize.	**detail extensive**
Transit statistics	Statistics for traffic transiting the router. When a burst of traffic is received, the value in the output packet rate field might briefly exceed the peak cell rate. This counter usually takes less than 1 second to stabilize. NOTE: The packet and byte counts in these fields include traffic that is dropped and does not leave the router.	**detail extensive**
Keepalive settings	(PPP and HDLC) Configured settings for keepalives. • **interval** ***seconds***—The time in seconds between successive keepalive requests. The range is **10** seconds through **32,767** seconds, with a default of **10** seconds. • **down-count*number***—The number of keepalive packets a destination must fail to receive before the network takes a link down. The range is **1** through **255**, with a default of **3**. • **up-count** ***number***—The number of keepalive packets a destination must receive to change a link's status from down to up. The range is **1** through **255**, with a default of **1**.	**detail extensive**

Keepalive statistics	(PPP and HDLC) Information about keepalive packets.
	
	• **Input**—Number of keepalive packets received by PPP.
	  ◦ **(last seen 00:00:00 ago)**—Time the last keepalive packet was received, in the format ***hh:mm:ss***.
	• **Output**—Number of keepalive packets sent by PPP and how long ago the last keepalive packets were sent and received.
	  ◦ **(last seen 00:00:00 ago)**—Time the last keepalive packet was sent, in the format ***hh:mm:ss***.
	
	(MX Series routers with MPCs/MICs) When an MX Series router with MPCs/MICs is using PPP fast keepalive for a PPP link, the display does not include the number of keepalive packets received or sent, or the amount of time since the router received or sent the last keepalive packet.
		detail extensive

Input packets	Number of packets received on the logical interface.	None specified
Output packets	Number of packets transmitted on the logical interface.	None specified

Table 17: show interfaces (PPPoE) Output Fields *(continued)*

Field Name	Field Description	Level of Output
LCP state	(PPP) Link Control Protocol state. • **Conf-ack-received**—Acknowledgement was received. • **Conf-ack-sent**—Acknowledgement was sent. • **Conf-req-sent**—Request was sent. • **Down**—LCP negotiation is incomplete (not yet completed or has failed). • **Not-configured**—LCP is not configured on the interface. • **Opened**—LCP negotiation is successful.	none **detail extensive**
NCP state	(PPP) Network Control Protocol state. • **Conf-ack-received**—Acknowledgement was received. • **Conf-ack-sent**—Acknowledgement was sent. • **Conf-req-sent**—Request was sent. • **Down**—NCP negotiation is incomplete (not yet completed or has failed). • **Not-configured**—NCP is not configured on the interface. • **Opened**—NCP negotiation is successful.	**detail extensive** none
CHAP state	(PPP) Displays the state of the Challenge Handshake Authentication Protocol (CHAP) during its transaction. • **Chap-Chal-received**—Challenge was received but response not yet sent. • **Chap-Chal-sent**—Challenge was sent. • **Chap-Resp-received**—Response was received for the challenge sent, but CHAP has not yet moved into the Success state. (Most likely with RADIUS authentication.) • **Chap-Resp-sent**—Response was sent for the challenge received. • **Closed**—CHAP authentication is incomplete. • **Failure**—CHAP authentication failed. • **Not-configured**—CHAP is not configured on the interface. • **Success**—CHAP authentication was successful.	none **detail extensive**
Protocol	Protocol family configured on the logical interface.	**detail extensive** none
protocol-family	Protocol family configured on the logical interface. If the protocol is **inet**, the IP address of the interface is also displayed.	**brief**
MTU	MTU size on the logical interface.	**detail extensive** none
Generation	Unique number for use by Juniper Networks technical support only.	**detail extensive**
Route table	Routing table in which the logical interface address is located. For example, **0** refers to the routing table inet.0.	**detail extensive** none
Flags	Information about the protocol family flags. Possible values are described in the "Family Flags" section under *Common Output Fields Description*.	**detail extensive** none

Table 17: show interfaces (PPPoE) Output Fields *(continued)*

Field Name	Field Description	Level of Output
Addresses, Flags	Information about the addresses configured for the protocol family. Possible values are described in the “Addresses Flags” section under *Common Output Fields Description*.	**detail extensive** none
Destination	IP address of the remote side of the connection.	**detail extensive** none
Local	IP address of the logical interface.	**detail extensive** none
Broadcast	Broadcast address.	**detail extensive** none

Sample Output

show interfaces (PPPoE)

```
user@host> show interfaces pp0
Physical interface: pp0, Enabled, Physical link is Up
  Interface index: 128, SNMP ifIndex: 24
  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 rate     : 0 bps (0 pps)
  Output rate    : 0 bps (0 pps)

  Logical interface pp0.0 (Index 72) (SNMP ifIndex 72)
    Flags: Hardware-Down Point-To-Point SNMP-Traps 0x4000 Encapsulation: PPPoE
    PPPoE:
      State: SessionDown, Session ID: None,
      Service name: None, Configured AC name: sapphire,
      Auto-reconnect timeout: 100 seconds, Idle timeout: Never,
      Underlying interface: at-5/0/0.0 (Index 70)
  Input packets : 0
  Output packets: 0
  LCP state: Not-configured
  NCP state: inet: Not-configured, inet6: Not-configured, iso: Not-configured,
  mpls: Not-configured
  CHAP state: Closed
    Protocol inet, MTU: 100
      Flags: User-MTU, Negotiate-Address
```

show interfaces (PPPoE over Aggregated Ethernet)

```
user@host> show interfaces pp0.1073773821
Logical interface pp0.1073773821 (Index 80) (SNMP ifIndex 32584)
    Flags: Point-To-Point SNMP-Traps 0x4000 Encapsulation: PPPoE
    PPPoE:
      State: SessionUp, Session ID: 1,
      Session AC name: alcor, Remote MAC address: 00:10:94:00:00:01,
      Underlying interface: demux0.100 (Index 88)
    Link:
      ge-1/0/0.32767
      ge-1/0/1.32767
    Input packets : 6
```

```
    Output packets: 6
  LCP state: Opened
  NCP state: inet: Opened, inet6: Not-configured, iso: Not-configured,  mpls:
Not-configured
  CHAP state: Closed
  PAP state: Success
    Protocol inet, MTU: 1500
      Flags: Sendbcast-pkt-to-re
      Addresses, Flags: Is-Primary
        Local: 45.63.24.1
```

show interfaces brief (PPPoE)

```
user@host> show interfaces pp0 brief
Physical interface: pp0, Enabled, Physical link is Up
  Type: PPPoE, Link-level type: PPPoE, MTU: 1532, Speed: Unspecified
  Device flags   : Present Running
  Interface flags: Point-To-Point SNMP-Traps

  Logical interface pp0.0
    Flags: Hardware-Down Point-To-Point SNMP-Traps 0x4000 Encapsulation: PPPoE
    PPPoE:
      State: SessionDown, Session ID: None,
      Service name: None, Configured AC name: sapphire,
      Auto-reconnect timeout: 100 seconds, Idle timeout: Never,
      Underlying interface: at-5/0/0.0 (Index 70)
    inet
```

show interfaces detail (PPPoE)

```
user@host> show interfaces pp0 detail
Physical interface: pp0, Enabled, Physical link is Up
  Interface index: 128, SNMP ifIndex: 24, Generation: 9
  Type: PPPoE, Link-level type: PPPoE, MTU: 1532, Speed: Unspecified
  Device flags   : Present Running
  Interface flags: Point-To-Point SNMP-Traps
  Link type      : Full-Duplex
  Link flags     : None
  Physical info  : Unspecified
  Hold-times     : Up 0 ms, Down 0 ms
  Current address: Unspecified, Hardware address: Unspecified
  Alternate link address: Unspecified
  Statistics last cleared: Never
  Traffic statistics:
   Input  bytes  :                    0                    0 bps
   Output bytes  :                    0                    0 bps
   Input  packets:                    0                    0 pps
   Output packets:                    0                    0 pps
  Logical interface pp0.0 (Index 72) (SNMP ifIndex 72) (Generation 14)
    Flags: Hardware-Down Point-To-Point SNMP-Traps 0x4000 Encapsulation: PPPoE
    PPPoE:
      State: SessionDown, Session ID: None,
      Service name: None, Configured AC name: sapphire,
      Auto-reconnect timeout: 100 seconds, Idle timeout: Never,
      Underlying interface: at-5/0/0.0 (Index 70)
    Traffic statistics:
     Input  bytes  :                    0
     Output bytes  :                    0
     Input  packets:                    0
     Output packets:                    0
    Local statistics:
```

```
    Input  bytes  :                   0
    Output bytes  :                   0
    Input  packets:                   0
    Output packets:                   0
   Transit statistics:
    Input  bytes  :                   0                    0 bps
    Output bytes  :                   0                    0 bps
    Input  packets:                   0                    0 pps
    Output packets:                   0                    0 pps
 LCP state: Not-configured
 NCP state: inet: Not-configured, inet6: Not-configured, iso: Not-configured,
 mpls: Not-configured
 CHAP state: Closed
   Protocol inet, MTU: 100, Generation: 14, Route table: 0
     Flags: User-MTU, Negotiate-Address
```

show interfaces detail (PPPoE on J Series Services Routers)

user@host> **show interfaces pp0 detail**

```
Physical interface: pp0, Enabled, Physical link is Up
  Interface index: 128, SNMP ifIndex: 24, Generation: 9
  Type: PPPoE, Link-level type: PPPoE, MTU: 1532, Speed: Unspecified
  Device flags   : Present Running
  Interface flags: Point-To-Point SNMP-Traps
  Link type      : Full-Duplex
  Link flags     : None
  Physical info  : Unspecified
  Hold-times     : Up 0 ms, Down 0 ms
  Current address: Unspecified, Hardware address: Unspecified
  Alternate link address: Unspecified
  Statistics last cleared: Never
  Traffic statistics:
   Input  bytes  :                   0                    0 bps
   Output bytes  :                   0                    0 bps
   Input  packets:                   0                    0 pps
   Output packets:                   0                    0 pps
  Input errors:
    Errors: 0, Drops: 0, Framing errors: 0, Runts: 0, Giants: 0,
    Policed discards: 0, Resource errors: 0
  Output errors:
    Carrier transitions: 0, Errors: 0, Drops: 0, MTU errors: 0,
    Resource errors: 0

  Logical interface pp0.0 (Index 72) (SNMP ifIndex 72) (Generation 14)
    Flags: Hardware-Down Point-To-Point SNMP-Traps 0x4000 Encapsulation: PPPoE
    PPPoE:
      State: SessionDown, Session ID: None,
      Service name: None, Configured AC name: sapphire,
      Auto-reconnect timeout: 100 seconds, Idle timeout: Never,
      Underlying interface: at-5/0/0.0 (Index 70)
    Traffic statistics:
     Input  bytes  :                   0
     Output bytes  :                   0
     Input  packets:                   0
     Output packets:                   0
    Local statistics:
     Input  bytes  :                   0
     Output bytes  :                   0
     Input  packets:                   0
     Output packets:                   0
    Transit statistics:
```

```
       Input  bytes  :                   0                    0 bps
       Output bytes  :                   0                    0 bps
       Input  packets:                   0                    0 pps
       Output packets:                   0                    0 pps
    LCP state: Not-configured
    NCP state: inet: Not-configured, inet6: Not-configured, iso: Not-configured,
    mpls: Not-configured
    CHAP state: Closed
      Protocol inet, MTU: 100, Generation: 14, Route table: 0
        Flags: User-MTU, Negotiate-Address
```

show interfaces extensive (PPPoE on M120 and M320 Routers)

```
user@host> show interfaces pp0 extensive
Physical interface: pp0, Enabled, Physical link is Up
  Interface index: 128, SNMP ifIndex: 93, Generation: 129
  Type: PPPoE, Link-level type: PPPoE, MTU: 1532, Speed: Unspecified
  Device flags   : Present Running
  Interface flags: Point-To-Point SNMP-Traps
  Link type      : Full-Duplex
  Link flags     : None
  Physical info  : Unspecified
  Hold-times     : Up 0 ms, Down 0 ms
  Current address: Unspecified, Hardware address: Unspecified
  Alternate link address: Unspecified
  Statistics last cleared: Never
  Traffic statistics:
   Input  bytes  :              972192                    0 bps
   Output bytes  :              975010                    0 bps
   Input  packets:                1338                    0 pps
   Output packets:                1473                    0 pps
   IPv6 transit statistics:
    Input  bytes  :                  0
    Output bytes  :                  0
    Input  packets:                  0
    Output packets:                  0
  Input errors:
    Errors: 0, Drops: 0, Framing errors: 0, Runts: 0, Giants: 0, Policed discards:
 0,
    Resource errors: 0
  Output errors:
    Carrier transitions: 0, Errors: 0, Drops: 0, MTU errors: 0, Resource errors:
 0

  Logical interface pp0.0 (Index 69) (SNMP ifIndex 96) (Generation 194)
    Flags: Point-To-Point SNMP-Traps 0x4000 Encapsulation: PPPoE
    PPPoE:
      State: SessionUp, Session ID: 26,
      Session AC name: None, AC MAC address: 00:17:cb:48:c8:12,
      Service name: None, Configured AC name: None,
      Auto-reconnect timeout: Never, Idle timeout: Never,
      Underlying interface: ge-3/0/1.0 (Index 67)
    Traffic statistics:
     Input  bytes  :                 252
     Output bytes  :                 296
     Input  packets:                   7
     Output packets:                   8
     IPv6 transit statistics:
      Input  bytes  :                  0
      Output bytes  :                  0
      Input  packets:                  0
```



```
      Output packets:                   0
    Local statistics:
     Input  bytes  :                  252
     Output bytes  :                  296
     Input  packets:                    7
     Output packets:                    8
    Transit statistics:
     Input  bytes  :                    0                     0 bps
     Output bytes  :                    0                     0 bps
     Input  packets:                    0                     0 pps
     Output packets:                    0                     0 pps
     IPv6 transit statistics:
      Input  bytes  :                   0
      Output bytes  :                   0
      Input  packets:                   0
      Output packets:                   0
  Keepalive settings: Interval 10 seconds, Up-count 1, Down-count 3
  Keepalive statistics:
    Input : 1 (last seen 00:00:00 ago)
    Output: 1 (last sent 00:00:03 ago)
  LCP state: Opened
  NCP state: inet: Opened, inet6: Not-configured, iso: Not-configured, mpls:
Not-configured
  CHAP state: Closed
  PAP state: Closed
    Protocol inet, MTU: 1492, Generation: 171, Route table: 0
      Flags: None
      Addresses, Flags: Is-Preferred Is-Primary
        Destination: 12.12.12.2, Local: 12.12.12.1, Broadcast: Unspecified,
Generation: 206
```


show interfaces demux0 (Demux Interfaces)

Syntax

show interfaces demux0.*logical-interface-number*
<brief | detail | extensive | terse>
<descriptions>
<media>
<snmp-index *snmp-index*>
<statistics>

Release Information

Command introduced in Junos OS Release 9.0.

Description

(MX Series and M Series routers only) Display status information about the specified demux interface.

Options

none—Display standard information about the specified demux interface.

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

descriptions—(Optional) Display interface description strings.

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

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

statistics—(Optional) Display static interface statistics.

Required Privilege Level

view

Related Documentation

- Verifying and Managing Agent Circuit Identifier-Based Dynamic VLAN Configuration on page 38

List of Sample Output

show interfaces (Demux) on page 712
show interfaces (PPPoE over Aggregated Ethernet) on page 713
show interfaces extensive (Targeted Distribution for Aggregated Ethernet Links) on page 714
show interfaces demux0 (ACI Interface Set Configured) on page 714

Output Fields

Table 18 on page 706 lists the output fields for the **show interfaces** (demux interfaces) command. Output fields are listed in the approximate order in which they appear.

Table 18: Demux show interfaces Output Fields

Field Name	Field Description	Level of Output
Physical Interface		
Physical interface	Name of the physical interface.	**brief detail extensive** none

Table 18: Demux show interfaces Output Fields *(continued)*

Field Name	Field Description	Level of Output
Interface index	Index number of the physical interface, which reflects its initialization sequence.	**brief detail extensive** none
Enabled	State of the interface. Possible values are described in the "Enabled Field" section under *Common Output Fields Description*.	**brief detail extensive** none
Physical link	Status of the physical link (**Up** or **Down**).	**detail extensive** none
Admin	Administrative state of the interface (**Up** or **Down**).	**terse**
Interface index	Index number of the physical interface, which reflects its initialization sequence.	**detail extensive** none
Link	Status of the physical link (**Up** or **Down**).	**terse**
Targeting summary	Status of aggregated Ethernet links that are configured with targeted distribution (**primary** or **backup**)	**extensive**
Bandwidth	Bandwidth allocated to the aggregated Ethernet links that are configured with targeted distribution.	**extensive**
Proto	Protocol family configured on the interface.	**terse**
SNMP ifIndex	SNMP index number for the physical interface.	**detail extensive** none
Generation	Unique number for use by Juniper Networks technical support only.	**detail extensive**
Type	Type of interface. **Software-Pseudo** indicates a standard software interface with no associated hardware device.	**brief detail extensive** none
Link-level type	Encapsulation being used on the physical interface.	**brief detail extensive**
MTU	Maximum transmission unit size on the physical interface.	**brief detail extensive**
Clocking	Reference clock source: **Internal** (1) or **External** (2).	**brief detail extensive**
Speed	Speed at which the interface is running.	**brief detail extensive**
Device flags	Information about the physical device. Possible values are described in the "Device Flags" section under *Common Output Fields Description*.	**brief detail extensive** none
Interface flags	Information about the interface. Possible values are described in the "Interface Flags" section under *Common Output Fields Description*.	**brief detail extensive** none
Link type	Data transmission type.	**detail extensive** none
Link flags	Information about the link. Possible values are described in the "Link Flags" section under *Common Output Fields Description*.	**detail extensive** none

Table 18: Demux show interfaces Output Fields *(continued)*

Field Name	Field Description	Level of Output
Physical info	Information about the physical interface.	**detail extensive**
Hold-times	Current interface hold-time up and hold-time down, in milliseconds.	**detail extensive**
Current address	Configured MAC address.	**detail extensive**
Hardware address	Hardware MAC address.	**detail extensive**
Alternate link address	Backup address of the link.	**detail extensive**
Last flapped	Date, time, and how long ago the interface went from down to up. The format is **Last flapped:** ***year-month-day hour:minute:second:timezone*** **(*hour:minute:second* ago)**. For example, **Last flapped: 2002-04-26 10:52:40 PDT (04:33:20 ago).**	**detail extensive** none
Statistics last cleared	Time when the statistics for the interface were last set to zero.	**detail extensive**
Traffic statistics	Number and rate of bytes and packets received and transmitted on the physical interface. • **Input bytes**—Number of bytes received on the interface. • **Output bytes**—Number of bytes transmitted on the interface. • **Input packets**—Number of packets received on the interface. • **Output packets**—Number of packets transmitted on the interface. • **IPv6 transit statistics**—Number of IPv6 transit bytes and packets received and transmitted on the physical interface if IPv6 statistics tracking is enabled. NOTE: These fields include dropped traffic and exception traffic, as those fields are not separately defined. ◦ **Input bytes**—Number of bytes received on the interface ◦ **Output bytes**—Number of bytes transmitted on the interface. ◦ **Input packets**—Number of packets received on the interface. ◦ **Output packets**—Number of packets transmitted on the interface.	**detail extensive**

Table 18: Demux show interfaces Output Fields *(continued)*

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

Table 18: Demux show interfaces Output Fields *(continued)*

Field Name	Field Description	Level of Output
ACI VLAN: Dynamic Profile	Name of the dynamic profile that defines the agent circuit identifier (ACI) interface set. If configured, the ACI interface set enables the underlying demux interface to create dynamic VLAN subscriber interfaces based on ACI information.	**brief detail extensive** none
Demux	Specific IP demultiplexing (demux) values: • **Underlying interface**—The underlying interface that the demux interface uses. • **Index**—Index number of the logical interface. • **Family**—Protocol family configured on the logical interface. • **Source prefixes, total**—Total number of source prefixes for the underlying interface. • **Destination prefixes, total**—Total number of destination prefixes for the underlying interface. • **Prefix**—**inet** family prefix.	**detail extensive** none
protocol-family	Protocol family configured on the logical interface.	**brief**
Traffic statistics	Number and rate of bytes and packets received and transmitted on the specified interface set. • **Input bytes, Output bytes**—Number of bytes received and transmitted on the interface set. • **Input packets, Output packets**—Number of packets received and transmitted on the interface set. • **IPv6 transit statistics**—Number of IPv6 transit bytes and packets received and transmitted on the logical interface if IPv6 statistics tracking is enabled. NOTE: The packet and byte counts in these fields include traffic that is dropped and does not leave the router. ◦ **Input bytes**—Number of bytes received on the interface. ◦ **Output bytes**—Number of bytes transmitted on the interface. ◦ **Input packets**—Number of packets received on the interface. ◦ **Output packets**—Number of packets transmitted on the interface.	**detail extensive**
Local statistics	Number of transit bytes and packets received and transmitted on the local interface. • **Input bytes**—Number of bytes received on the interface. • **Output bytes**—Number of bytes transmitted on the interface. • **Input packets**—Number of packets received on the interface. • **Output packets**—Number of packets transmitted on the interface.	**detail extensive**

Table 18: Demux show interfaces Output Fields *(continued)*

Field Name	Field Description	Level of Output
Transit statistics	Number and rate of bytes and packets transiting the switch. NOTE: The packet and byte counts in these fields include traffic that is dropped and does not leave the router. • **Input bytes**—Number of bytes received on the interface. • **Output bytes**—Number of bytes transmitted on the interface. • **Input packets**—Number of packets received on the interface. • **Output packets**—Number of packets transmitted on the interface.	**detail extensive**
IPv6 Transit statistics	Number of IPv6 transit bytes and packets received and transmitted on the logical interface if IPv6 statistics tracking is enabled. NOTE: The packet and byte counts in these fields include traffic that is dropped and does not leave the router. • **Input bytes**—Number of bytes received on the interface. • **Output bytes**—Number of bytes transmitted on the interface. • **Input packets**—Number of packets received on the interface. • **Output packets**—Number of packets transmitted on the interface.	**detail extensive**
Input packets	Number of packets received on the interface.	none
Output packets	Number of packets transmitted on the interface.	none
Protocol	Protocol family. Possible values are described in the “Protocol Field” section under *Common Output Fields Description*.	**detail extensive** none
MTU	Maximum transmission unit size on the logical interface.	**detail extensive** none
Maximum labels	Maximum number of MPLS labels configured for the MPLS protocol family on the logical interface.	**detail extensive** none
Generation	Unique number for use by Juniper Networks technical support only.	**detail extensive**
Route table	Route table in which the logical interface address is located. For example, **0** refers to the routing table inet.0.	**detail extensive**
Flags	Information about protocol family flags. Possible values are described in the “Family Flags” section under *Common Output Fields Description*.	**detail extensive** none
Mac-Validate Failures	Number of MAC address validation failures for packets and bytes. This field is displayed when MAC address validation is enabled for the logical interface.	**detail extensive** none
Addresses, Flags	Information about the address flags. Possible values are described in the “Addresses Flags” section under *Common Output Fields Description*.	**detail extensive** none
Destination	IP address of the remote side of the connection.	**detail extensive statistics** none

Table 18: Demux show interfaces Output Fields *(continued)*

Field Name	Field Description	Level of Output
Local	IP address of the logical interface.	**detail extensive terse** none
Remote	IP address of the remote interface.	**terse**
Broadcast	Broadcast address of the logical interlace.	**detail extensive** none
Generation	Unique number for use by Juniper Networks technical support only.	**detail extensive**
Link	Name of the physical interfaces for member links in an aggregated Ethernet bundle for a PPPoE over aggregated Ethernet configuration. PPPoE traffic goes out on these interfaces.	**detail extensive** none
Dynamic-profile	Name of the PPPoE dynamic profile assigned to the underlying interface.	**detail extensive** none
Service Name Table	Name of the PPPoE service name table assigned to the PPPoE underlying interface.	**detail extensive** none
Max Sessions	Maximum number of dynamic PPPoE logical interfaces that the router can activate on the underlying interface.	**detail extensive** none
Duplicate Protection	State of duplicate protection: **On** or **Off**. Duplicate protection prevents the activation of another dynamic PPPoE logical interface on the same underlying interface when a dynamic PPPoE logical interface for a client with the same MAC address is already active on that interface.	**detail extensive** none
Direct Connect	State of the configuration to ignore DSL Forum VSAs: **On** or **Off**. When configured, the router ignores any of these VSAs received from a directly connected CPE device on the interface.	**detail extensive** none
AC Name	Name of the access concentrator.	**detail extensive** none

Sample Output

show interfaces (Demux)

```
user@host> show interfaces demux0
Physical interface: demux0, Enabled, Physical link is Up
  Interface index: 128, SNMP ifIndex: 79, Generation: 129
  Type: Software-Pseudo, Link-level type: Unspecified, MTU: 9192, Clocking: 1,
  Speed: Unspecified
  Device flags   : Present Running
  Interface flags: Point-To-Point SNMP-Traps
  Link type      : Full-Duplex
  Link flags     : None
  Physical info  : Unspecified
  Hold-times     : Up 0 ms, Down 0 ms
  Current address: Unspecified, Hardware address: Unspecified
  Alternate link address: Unspecified
  Last flapped   : Never
  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
   IPv6 transit statistics:
    Input  bytes  :                   0
    Output bytes  :                   0
    Input  packets:                   0
    Output packets:                   0
  Input errors:
    Errors: 0, Drops: 0, Framing errors: 0, Runts: 0, Giants: 0,
    Policed discards: 0, Resource errors: 0
  Output errors:
    Carrier transitions: 0, Errors: 0, Drops: 0, MTU errors: 0,
    Resource errors: 0

Logical interface demux0.0 (Index 87) (SNMP ifIndex 84) (Generation 312)
    Flags: SNMP-Traps 0x4000 Encapsulation: ENET2
    Demux:
     Underlying interface: ge-2/0/1.0 (Index 74)
   Family Inet Source prefixes, total 1
   Prefix: 1.1.1/24
    Traffic statistics:
     Input  bytes  :                   0
     Output bytes  :                1554
     Input  packets:                   0
     Output packets:                  37
     IPv6 transit statistics:
      Input  bytes  :                  0
      Output bytes  :                  0
      Input  packets:                  0
      Output packets:                  0
    Local statistics:
     Input  bytes  :                   0
     Output bytes  :                1554
     Input  packets:                   0
     Output packets:                  37
    Transit statistics:
     Input  bytes  :                   0                    0 bps
     Output bytes  :                   0                    0 bps
     Input  packets:                   0                    0 pps
     Output packets:                   0                    0 pps
     IPv6 transit statistics:
      Input  bytes  :                  0
      Output bytes  :                  0
      Input  packets:                  0
      Output packets:                  0
    Protocol inet, MTU: 1500, Generation: 395, Route table: 0
      Flags: Is-Primary, Mac-Validate-Strict
      Mac-Validate Failures: Packets: 0, Bytes: 0
      Addresses, Flags: Is-Preferred Is-Primary
        Destination: 11.1.1/24, Local: 11.1.1.1, Broadcast: 11.1.1.255,
        Generation: 434
```

show interfaces (PPPoE over Aggregated Ethernet)

```
user@host> show interfaces demux0.100
Logical interface demux0.100 (Index 76) (SNMP ifIndex 61160)
    Flags: SNMP-Traps 0x4000 VLAN-Tag [ 0x8100.100 ]
    Encapsulation: ENET2
```



```
    Demux:
      Underlying interface: ae0 (Index 199)
    Link:
      ge-1/0/0
      ge-1/1/0
    Input packets : 0
    Output packets: 0
    Protocol pppoe
      Dynamic Profile: pppoe-profile,
      Service Name Table: service-table1,
      Max Sessions: 100, Duplicate Protection: On,
      Direct Connect: Off,
      AC Name: pppoe-server-1
```

show interfaces extensive (Targeted Distribution for Aggregated Ethernet Links)

user@host> **show interfaces demux0.1073741824 extensive**

```
  Logical interface demux0.1073741824 (Index 75) (SNMP ifIndex 558) (Generation
346)
    Flags: SNMP-Traps 0x4000 VLAN-Tag [ 0x8100.1 ]  Encapsulation: ENET2
    Demux:
      Underlying interface: ae0 (Index 201)
    Link:
      ge-1/0/0
      ge-1/1/0
      ge-2/0/7
      ge-2/0/8
    Targeting summary:
      ge-1/1/0, primary, Physical link is Up
      ge-2/0/8, backup, Physical link is Up
    Bandwidth: 1000mbps
```

show interfaces demux0 (ACI Interface Set Configured)

user@host> **show interfaces demux0.1073741827**

```
  Logical interface demux0.1073741827 (Index 346) (SNMP ifIndex 527)
    Flags: SNMP-Traps 0x4000 VLAN-Tag [ 0x8100.1802 0x8100.302 ]  Encapsulation:
 ENET2
    Demux: Source Family Inet
    ACI VLAN:
      Dynamic Profile: aci-vlan-set-profile
    Demux:
      Underlying interface: ge-1/0/0 (Index 138)
    Input packets : 18
    Output packets: 16
    Protocol inet, MTU: 1500
      Flags: Sendbcast-pkt-to-re, Unnumbered
      Donor interface: lo0.0 (Index 322)
      Preferred source address: 100.20.200.202
      Addresses, Flags: Primary Is-Default Is-Primary
        Local: 10.4.12.119
    Protocol pppoe
      Dynamic Profile: aci-vlan-pppoe-profile,
      Service Name Table: None,
      Max Sessions: 32000, Max Sessions VSA Ignore: Off,
      Duplicate Protection: On, Short Cycle Protection: Off,
      Direct Connect: Off,
      AC Name: nbc
```


show interfaces interface-set (Ethernet Interface Set)

Syntax

show interfaces interface-set *interface-set-name*
<detail | terse>

Release Information

Command introduced in Junos OS Release 8.5.

Description

Display information about the specified gigabit or 10-Gigabit Ethernet interface set. Supported in MX Series routers with enhanced queuing DPCs or MPCs.

You can also use the **show interfaces interface-set** command to display information about agent circuit identifier (ACI) interface sets configured on MX Series routers with MPCs/MICs.

Options

interface-set ***interface-set-name***—Display information about the specified Gigabit Ethernet, 10-Gigabit Ethernet, or ACI interface set.

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

Required Privilege Level

view

Related Documentation

- Verifying and Managing Agent Circuit Identifier-Based Dynamic VLAN Configuration on page 38

List of Sample Output

show interfaces interface-set terse on page 716
show interfaces interface-set detail on page 716
show interfaces interface-set (ACI Interface Set) on page 717

Output Fields

Table 19 on page 715 describes the information for the **show interfaces interface-set** command.

Table 19: Ethernet show interfaces interface-set Output Fields

Field Name	Field Description	Level of Output
Physical Interface		
Interface set	Name of the interface set or sets.	All levels
Interface set index	Index number of the interface set. For ACI interface sets, the following fields are displayed: • **ACI VLAN**—ACI interface set that the router uses to create dynamic VLAN subscriber interfaces based on the agent circuit identifier value. • **PPPoE**—Dynamic PPPoE subscriber interface that the router creates using the ACI interface set.	**detail** none
Agent Circuit ID	For ACI interface sets, string in DHCP or PPPoE control packets that uniquely identifies the subscriber's access node and and the DSL line on the access node.	**detail** none
Max Sessions	For dynamic PPPoE subscriber interfaces, maximum number of PPPoE logical interfaces that that can be activated on the underlying interface.	**detail** none

Table 19: Ethernet show interfaces interface-set Output Fields *(continued)*

Field Name	Field Description	Level of Output
Max Sessions VSA Ignore	For dynamic PPPoE subscriber interfaces, whether the router is configured to ignore (clear) the PPPoE maximum session value returned by RADIUS in the Max-Clients-Per-Interface Juniper Networks VSA [26-143] and restore the PPPoE maximum session value on the underlying interface to the value configured with the **max-sessions** statement: **Off** (default) or **On**.	**detail** none
Traffic statistics	Number and rate of bytes and packets received and transmitted on the specified interface set. • **Input bytes, Output bytes**—Number of bytes and number of bytes per second received and transmitted on the interface set • **Input packets, Output packets**—Number of packets and number of packets per second received and transmitted on the interface set.	**detail**
Egress queues supported	Total number of egress queues supported on the specified interface set.	**detail**
Egress queues in use	Total number of egress queues used on the specified interface set.	**detail**
Queue counters	**Queued packets**, **Transmitted packets**, and **Dropped packets** statistics for the four forwarding classes.	**detail**
Members	List of all interface sets or, for ACI interface sets, list of all subscriber interfaces belonging to the specified ACI interface set.	**detail** none

Sample Output

show interfaces interface-set terse

```
user@host> show interfaces interface-set terse
Interface set:
   iflset-xe-11/3/0-0
   ge-1/0/1-0
   ge-1/0/1-2
```

show interfaces interface-set detail

```
user@host> show interfaces interface-set iflset-xe-11/3/0-0 detail
Interface set: iflset-xe-11/3/0-0
  Interface set index: 19
  Traffic statistics:
   Output bytes  :            751017840            401673504 bps
   Output packets:             11044380               738377 pps
  Egress queues: 4 supported, 4 in use
  Queue counters:       Queued packets  Transmitted packets    Dropped packets
    0 best-effort            211091327             11044380          199995746
    1 expedited-fo                   0                    0                  0
    2 assured-forw                   0                    0                  0
    3 network-cont                   0                    0                  0
  Members:
   xe-11/3/0.0
```

show interfaces interface-set (ACI Interface Set)

```
user@host> show interfaces interface-set
Interface set: aci-1001-demux0.1073741826
  Interface set index: 1
    ACI VLAN:
      Agent Circuit ID: aci-ppp-dhcp-dvlan-60
    PPPoE:
      Max Sessions: 3, Max Sessions VSA Ignore: Off
  Members:
   pp0.1073741827
```

show interfaces ps0 (Pseudowire Subscriber Interfaces)

Syntax

show interfaces ps0
<brief | detail | extensive | terse>

Release Information Command introduced at Junos OS Release 13.1.

Description Display status information about the pseudowire subscriber interface.

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

Required Privilege Level view

Related Documentation

- Pseudowire Subscriber Logical Interfaces Overview on page 321

List of Sample Output show interfaces ps0 on page 720

Output Fields Table 17 on page 695 lists the output fields for the **show interfaces ps0** command. Output fields are listed in the approximate order in which they appear.

Table 20: show interfaces ps0 Output Fields

Field Name	Field Description	Level of Output
Physical Interface		
Physical interface	Name of the physical interface.	**brief detail extensive** none
Enabled	State of the interface. Possible values are described in the “Enabled Field” section under *Common Output Fields Description*.	**brief detail extensive** none
Interface index	Physical interface index number, which reflects its initialization sequence.	**detail extensive** none
SNMP ifIndex	SNMP index number for the physical interface.	**detail extensive** none
Type	Physical interface type (Software-Pseudo).	**brief detail extensive** none
Link-level type	Encapsulation being used on the physical interface.	**brief detail extensive**
MTU	MTU size on the physical interface.	**brief detail extensive**
Clocking	Reference clock source. It can be **Internal** or **External**.	**brief detail extensive**
Speed	Speed at which the interface is running.	**brief detail extensive**
Device flags	Information about the physical device. Possible values are described in the “Device Flags” section under *Common Output Fields Description*.	**brief detail extensive** none

Table 20: show interfaces ps0 Output Fields *(continued)*

Field Name	Field Description	Level of Output
Interface flags	Information about the interface. Possible values are described in the "Interface Flags" section under *Common Output Fields Description*.	**brief detail extensive** none
Current address	Configured MAC address.	**detail extensive** none
Hardware address	MAC address of the hardware.	**detail extensive** none
Last flapped	Date, time, and how long ago the interface went from down to up or up to down. The format is Last flapped: *year-month-day hours:minutes:seconds*: *timezone* (*hours:minutes:seconds* ago). or Never. For example, Last flapped: 2002-04-26 10:52:40 PDT (04:33:20 ago).	**detail extensive** none
input packets	Number of packets received on the logical interface.	**detail extensive** none
output packets	Number of packets transmitted on the logical interface.	**detail extensive** none
Logical Interface		
Logical interface	Name of the logical interface.	**brief detail extensive** none
Index	Logical interface index number (which reflects its initialization sequence).	**detail extensive** none
SNMP ifIndex	Logical interface SNMP interface index number.	**detail extensive** none
Flags	Information about the logical interface. Possible values are described in the "Logical Interface Flags" section under *Common Output Fields Description*.	**brief detail extensive** none
Encapsulation	Type of encapsulation configured on the logical interface.	**brief extensive** none
Input packets	Number of packets received on the logical interface.	none
Output packets	Number of packets transmitted on the logical interface.	none
Protocol	Protocol family configured on the logical interface.	**detail extensive** none
MTU	MTU size on the logical interface.	**detail extensive** none
Flags	Information about the protocol family flags. Possible values are described in the "Family Flags" section under *Common Output Fields Description*.	**detail extensive** none
Addresses, Flags	Information about the addresses configured for the protocol family. Possible values are described in the "Addresses Flags" section under *Common Output Fields Description*.	**detail extensive** none
Destination	IP address of the remote side of the connection.	**detail extensive** none
Local	IP address of the logical interface.	**detail extensive terse** none

Table 20: show interfaces ps0 Output Fields *(continued)*

Field Name	Field Description	Level of Output
Broadcast	Broadcast address.	**detail extensive** none

Sample Output

show interfaces ps0

```
user@host> show interfaces ps0
Physical interface: ps0, Enabled, Physical link is Up
 Interface index: 166, SNMP ifIndex: 658
 Type: Software-Pseudo, Link-level type: 90, MTU: 1518, Clocking: 1, Speed: 800mbps

 Device flags : Present Running
 Interface flags: Point-To-Point Internal: 0x4000
 Current address: 00:1d:b5:a8:19:4a, Hardware address: 00:1d:b5:a8:19:4a
 Last flapped : Never
  Input packets : 0
  Output packets: 0

 Logical interface ps0.0 (Index 74) (SNMP ifIndex 656)
  Flags: Point-To-Point 0x4000 Encapsulation: Ethernet-CCC
  Input packets : 482
  Output packets: 0
  Protocol ccc, MTU: 1518
   Flags: Is-Primary

 Logical interface ps0.1 (Index 78) (SNMP ifIndex 665)
  Flags: Point-To-Point 0x4000 VLAN-Tag [ 0x8100.100 ] Encapsulation: ENET2
  Input packets : 0
  Output packets: 482
  Protocol inet, MTU: 1500
   Flags: Sendbcast-pkt-to-re
    Addresses, Flags: Is-Preferred Is-Primary
     Destination: 20.0.0/24, Local: 20.0.0.1, Broadcast: 20.0.0.255

 Logical interface ps0.32767 (Index 75) (SNMP ifIndex 692)
  Flags: Point-To-Point 0x4000 VLAN-Tag [ 0x0000.0 ] Encapsulation: ENET2
  Input packets : 0
  Output packets: 0
```

show ppp interface

Syntax

show ppp interface *interface-name*
<extensive| terse>

Release Information

Command introduced in Junos OS Release 7.5.

Description

Display information about PPP interfaces.

Options

interface-name—Name of a logical interface.

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

Required Privilege Level

view

List of Sample Output

show ppp interface on page 729
show ppp interface extensive on page 729
show ppp interface terse on page 729

Output Fields

Table 21 on page 721 lists the output fields for the **show ppp interface** command. Output fields are listed in the approximate order in which they appear.

Table 21: show ppp interface Output Fields

Field Name	Field Description	Level of Output
Session	Name of the logical interface on which the session is running.	All levels
Type	Session type: PPP.	All levels
Phase	PPP process phase: **Authenticate**, **Pending**, **Establish**, **LCP**, **Network**, **Disabled**, and **Tunneled**.	All levels
Session flags	Special conditions present in the session: **Bundled**, **TCC**, **No-keepalives**, **Looped**, **Monitored**, and **NCP-only**.	All levels
***protocol* State**	Protocol state information. See specific protocol state fields for information.	None specified
AUTHENTICATION	Challenge-Handshake Authentication Protocol (CHAP) authentication state information or Password Authentication Protocol (PAP) state information. See the **Authentication** field description for further information.	None specified

Table 21: show ppp interface Output Fields *(continued)*

Field Name	Field Description	Level of Output
Keepalive settings	Keepalive settings for the PPP sessions on the L2TP network server (LNS). LNS based PPP sessions are supported only on service interfaces (si). • **Interval**—Time in seconds between successive keepalive requests. Keepalive aging timeout is calculated as a product of the **interval** and **Down-count** values. If the keepalive aging timeout is greater than 180 seconds, the keepalive packets are handled by the Routing Engine. If the aging timeout is less than or equal to 180 seconds, the packets are handled by the Packet Forwarding Engine. • **Up-count**—The number of keepalive packets a destination must receive to change a link's status from down to up. • **Down-count**—The number of keepalive packets a destination must fail to receive before the network takes down a link.	**extensive**
RE Keepalive statistics	Keepalive statistics for the packets handled by the Routing Engine. • **LCP echo req Tx**—LCP echo requests sent from the Routing Engine. • **LCP echo req Rx**—LCP echo requests received at the Routing Engine. • **LCP echo rep Tx**—LCP echo responses sent from the Routing Engine. • **LCP echo rep Rx**—LCP echo responses received at the Routing Engine. • **LCP echo req timeout**—Number of keepalive packets where the keepalive aging timer has expired. • **LCP Rx echo req Magic Num Failures**—LCP echo requests where the magic numbers shared between the PPP peers during LCP negotiation did not match. • **LCP Rx echo rep Magic Num Failures**—LCP echo responses where the magic numbers shared between the PPP peers during LCP negotiation did not match.	**extensive**

Table 21: show ppp interface Output Fields *(continued)*

Field Name	Field Description	Level of Output

LCP	**LCP information:**
	• **State**—LCP protocol state (all platforms except M120 and M320 routers):
	  ◦ **Ack-rcvd**—A Configure-Request has been sent and a Configure-Ack has been received.
	  ◦ **Ack-sent**—A Configure-Request and a Configure-Ack have both been sent, but a Configure-Ack has not yet been received.
	  ◦ **Closed**—Link is not available for traffic.
	  ◦ **Opened**—Link is administratively available for traffic.
	  ◦ **Req-sent**—An attempt has been made to configure the connection.
	• **State**—LCP protocol state (M120 and M320 routers):
	  ◦ **Ack-rcvd**—A Configure-Request has been sent and a Configure-Ack has been received.
	  ◦ **Ack-sent**—A Configure-Request and a Configure-Ack have both been sent, but a Configure-Ack has not yet been received.
	  ◦ **Closed**—Link is available (up), but no Open has occurred.
	  ◦ **Closing**—A Terminate-Request has been sent but a Terminate-Ack has not yet been received.
	  ◦ **Opened**—Link is administratively available for traffic. A Configure-Ack has been both sent and received.
	  ◦ **Req-sent**—An attempt has been made to configure the connection. A Configure-Request has been sent but a Configure-Ack has not yet been received.
	  ◦ **Starting**—An administrative Open has been initiated, but the lower layer is still unavailable (Down).
	  ◦ **Stopped**—The system is waiting for a Down event after the This-Layer-Finished action, or after sending a Terminate-Ack.
	  ◦ **Stopping**—A Terminate-Request has been sent but a Terminate-Ack has not yet been received.
	• **Last started**—LCP state start time.
	• **Last completed**—LCP state completion time.
		extensive

Table 21: show ppp interface Output Fields *(continued)*

Field Name	Field Description	Level of Output
	• **Negotiated options**: ◦ **ACFC**—Address and-Control Field Compression. A configuration option that provides a method to negotiate the compression of the Data Link Layer Address and Control fields. ◦ **Asynchronous map**—Asynchronous control character map. A configuration option used on asynchronous links such as telephone lines to identify control characters that must be replaced by a two-character sequence to prevent them from being interpreted by equipment used to establish the link. ◦ **Authentication protocol**—Protocol used for authentication. This option provides a method to negotiate the use of a specific protocol for authentication. It requires a peer to authenticate itself before allowing network-layer protocol packets to be exchanged. By default, authentication is not required. ◦ **Authentication algorithm**—Type of authentication algorithm. The Message Digest algorithm (MD5) is the only algorithm supported. ◦ **Endpoint discriminator class**—For multilink PPP (MLPPP), a configuration option that identifies the system transmitting the packet. This option advises a system that the peer on this link could be the same as the peer on another existing link. ◦ **Magic number**—A configuration option that provides a method to detect looped-back links and other data-link layer anomalies. By default, the magic number is not negotiated. ◦ **MRU**—Maximum receive unit. A configuration option that may be sent to inform the peer that the implementation can receive larger packets, or to request that the peer send smaller packets. The default value is 1500 octets. ◦ **MRRU**—For multilink PPP, the maximum receive reconstructed unit. A configuration option that specifies the maximum number of octets in the Information fields of reassembled packets. ◦ **Multilink header suspendable classes**—For MLPPP, an LCP option that advises the peer that the implementation wishes to receive fragments with a format given by the code number, with the maximum number of suspendable classes given. ◦ **Multilink header format classes**—For MLPPP, an LCP option that advises the peer that the implementation wishes to receive fragments with a format given by the code number. ◦ **PFC**—Protocol-Field-Compression. A configuration option that provides a method to negotiate the compression of the PPP Protocol field. ◦ **short sequence**—For MLPPP, an option that advises the peer that the implementation wishes to receive fragments with short, 12-bit sequence numbers.	

Table 21: show ppp interface Output Fields *(continued)*

Field Name	Field Description	Level of Output
Authentication	CHAP or PAP authentication state information. For CHAP authentication: • **Chap-ans-rcvd**—Packet was sent from the peer, indicating that the peer received the **Chap-resp-sent** packet. • **Chap-ans-sent**—Packet was sent from the authenticator, indicating that the authenticator received the peer's **Chap-resp-rcvd** packet. • **Chap-chal-rcvd**—Challenge packet has been received by the peer. • **Chap-chal-sent**—Challenge packet has been sent by the authenticator to begin the CHAP protocol or has been transmitted at any time during the Network-Layer Protocol (NCP) phase to ensure that the connection has not been altered. • **Chap-resp-rcvd**—CHAP response packet has been received by the authenticator. • **Chap-resp-sent**—CHAP response packet has been sent to the authenticator. • **Closed**—Link is not available for authentication. • **Failure**—Authenticator compares the response value in the response packet from the peer with its own response value, but the value does not match. Authentication fails. • **Success**—Authenticator compares the response value in the response packet from the peer with its own response value, and the value matches. Authentication is successful. For PAP authentication: • **Pap-resp-sent**—PAP response sent to peer (ACK/NACK)t. • **Pap-req-rcvd**—PAP request packet received from peer. • **Pap-resp-rcvd**—PAP response received from the peer (ACK/NACK). • **Pap-req-sent**—PAP request packet sent to the peer. • **Closed**—Link is not available for authentication. • **Failure**—Authenticator compares the response value in the response packet from the peer with its own response value, but the value does not match. Authentication fails. • **Success**—Authenticator compares the response value in the response packet from the peer with its own response value, and the value matches. Authentication is successful.	None specified

Table 21: show ppp interface Output Fields *(continued)*

Field Name	Field Description	Level of Output

IPCP	Internet Protocol Control Protocol (IPCP) information.
	• **State**—(All platforms except M120 and M320 routers) One of the following values:
	  ◦ **Ack-rcvd**—A Configure-Request has been sent and a Configure-Ack has been received.
	  ◦ **Ack-sent**—A Configure-Request and a Configure-Ack have both been sent, but a Configure-Ack has not yet been received.
	  ◦ **Closed**—Link is not available for traffic.
	  ◦ **Opened**—Link is administratively available for traffic.
	  ◦ **Req-sent**—An attempt has been made to configure the connection.
	• **State**—(M120 and M320 routers) One of the following values:
	  ◦ **Ack-rcvd**—A Configure-Request has been sent and a Configure-Ack has been received.
	  ◦ **Ack-sent**—A Configure-Request and a Configure-Ack have both been sent, but a Configure-Ack has not yet been received.
	  ◦ **Closed**—Link is available (up), but no Open has occurred.
	  ◦ **Closing**—A Terminate-Request has been sent but a Terminate-Ack has not yet been received.
	  ◦ **Opened**—Link is administratively available for traffic. A Configure-Ack has been both sent and received.
	  ◦ **Req-sent**—An attempt has been made to configure the connection. A Configure-Request has been sent but a Configure-Ack has not yet been received.
	  ◦ **Starting**—An administrative Open has been initiated, but the lower layer is still unavailable (Down).
	  ◦ **Stopped**—The system is waiting for a Down event after the This-Layer-Finished action, or after sending a Terminate-Ack.
	  ◦ **Stopping**—A Terminate-Request has been sent but a Terminate-Ack has not yet been received.
	• **Last started**—IPCP state start time.
	• **Last completed**—IPCP state authentication completion time.
	• **Negotiated options:**
	  ◦ **compression protocol**—Negotiate the use of a specific compression protocol. By default, compression is not enabled.
	  ◦ **local address**—Desired local address of the sender of a Configure-Request. If all four octets are set to zero, the peer provides the IP address.
	  ◦ **primary DNS server**—Negotiate with the remote peer to select the address of the primary DNS server to be used on the local end of the link.
	  ◦ **primary WINS server**—Negotiate with the remote peer to select the address of the primary WINS server to be used on the local end of the link.
	  ◦ **remote address**—IP address of the remote end of the link in dotted quad notation.
	  ◦ **secondary DNS server**—Negotiate with the remote peer to select the address of the secondary DNS server to be used on the local end of the link.
	  ◦ **secondary WINS server**—Negotiate with the remote peer to select the address of the secondary WINS server to be used on the local end of the link.
	• **Negotiation mode**—PPP Network Control Protocol (NCP) negotiation mode configured for IPCP: **Active** or **Passive**
		extensive

Table 21: show ppp interface Output Fields *(continued)*

<table>
<tr><th>Field Name</th><th>Field Description</th><th>Level of Output</th></tr>
<tr><td>IPV6CP</td><td>Internet Protocol version 6 Control Protocol (IPv6CP) information.

State—(All platforms except M120 and M320 routers) One of the following values:

Ack-rcvd—A Configure-Request has been sent and a Configure-Ack has been received.
Ack-sent—A Configure-Request and a Configure-Ack have both been sent, but a Configure-Ack has not yet been received.
Closed—Link is not available for traffic.
Opened—Link is administratively available for traffic.
Req-sent—An attempt has been made to configure the connection.

State—(M120 and M320 routers) One of the following values:

Ack-rcvd—A Configure-Request has been sent and a Configure-Ack has been received.
Ack-sent—A Configure-Request and a Configure-Ack have both been sent, but a Configure-Ack has not yet been received.
Closed—Link is available (up), but no Open has occurred.
Closing—A Terminate-Request has been sent but a Terminate-Ack has not yet been received.
Opened—Link is administratively available for traffic. A Configure-Ack has been both sent and received.
Req-sent—An attempt has been made to configure the connection. A Configure-Request has been sent but a Configure-Ack has not yet been received.
Starting—An administrative Open has been initiated, but the lower layer is still unavailable (Down).
Stopped—The system is waiting for a Down event after the This-Layer-Finished action, or after sending a Terminate-Ack.
Stopping—A Terminate-Request has been sent but a Terminate-Ack has not yet been received.

Last started—IPV6CP state start time.
Last completed—IPV6CP state authentication completion time.
Negotiated options:

local interface identifier—Desired local address of the sender of a Configure-Request. If all four octets are set to zero, the peer provides the IP address.
remote interface identifier—IP address of the remote end of the link in dotted quad notation.

Negotiation mode—PPP Network Control Protocol (NCP) negotiation mode configured for IPv6CP: Active or Passive
</td><td>extensive</td></tr>
</table>

Table 21: show ppp interface Output Fields *(continued)*

<table>
<tr><th>Field Name</th><th>Field Description</th><th>Level of Output</th></tr>
<tr><td>OSINLCP State</td><td>OSI Network Layer Control Protocol (OSINLCP) protocol state information (all platforms except M120 and M320 routers):
State:
Ack-rcvd—Configure-Request has been sent and Configure-Ack has been received.
Ack-sent—Configure-Request and Configure-Ack have both been sent, but Configure-Ack has not yet been received.
Closed—Link is not available for traffic.
Opened—Link is administratively available for traffic.
Req-sent—Attempt has been made to configure the connection.
Last started—OSINLCP state start time.
Last completed—OSINCLP state completion time.</td><td>extensive</td></tr>
<tr><td>TAGCP</td><td>TAGCP information.
State—(All platforms except M120 and M320 routers) One of the following values:
Ack-rcvd—A Configure-Request has been sent and a Configure-Ack has been received.
Ack-sent—A Configure-Request and a Configure-Ack have both been sent, but a Configure-Ack has not yet been received.
Closed—Link is not available for traffic.
Opened—Link is administratively available for traffic.
Req-sent—An attempt has been made to configure the connection.
State—(M120 and M320 routers) One of the following values:
Ack-rcvd—A Configure-Request has been sent and a Configure-Ack has been received.
Ack-sent—A Configure-Request and a Configure-Ack have both been sent, but a Configure-Ack has not yet been received.
Closed—Link is available (up), but no Open has occurred.
Closing—A Terminate-Request has been sent but a Terminate-Ack has not yet been received.
Opened—Link is administratively available for traffic. A Configure-Ack has been both sent and received.
Req-sent—An attempt has been made to configure the connection. A Configure-Request has been sent but a Configure-Ack has not yet been received.
Starting—An administrative Open has been initiated, but the lower layer is still unavailable (Down).
Stopped—The system is waiting for a Down event after the This-Layer-Finished action, or after sending a Terminate-Ack.
Stopping—A Terminate-Request has been sent but a Terminate-Ack has not yet been received.
Last started—TAGCP state start time.
Last completed—TAGCP state authentication completion time.</td><td>extensive
none</td></tr>
</table>

Sample Output

show ppp interface

```
user@host> show ppp interface si-1/3/0.0
Session si-1/3/0.0, Type: PPP, Phase: Authenticate
  Session flags: Monitored
    LCP State: Opened
    AUTHENTICATION: CHAP State: Chap-resp-sent, Chap-ans-sent
    IPCP State: Closed, OSINLCP State: Closed
```

show ppp interface extensive

```
user@host> show ppp interface si-0/0/3.0 extensive

    Session si-0/0/3.0, Type: PPP, Phase: Network
        Keepalive settings: Interval 30 seconds, Up-count 1, Down-count 3
RE Keepalive statistics:
  LCP echo req Tx        :  657 (last sent 00:50:10 ago)
  LCP echo req Rx        :    0 (last seen: never)
  LCP echo rep Tx        :    0
  LCP echo rep Rx        :  657
  LCP echo req timout    :    0
  LCP Rx echo req Magic Num Failures :    0
  LCP Rx echo rep Magic Num Failures :    0
LCP
  State: Opened
  Last started: 2007-01-29 10:43:50 PST
  Last completed: 2007-01-29 10:43:50 PST
  Negotiated options:
    Authentication protocol: PAP, Magic number: 2341124815, MRU: 4470
Authentication: PAP
  State: Success
  Last started: 2007-01-29 10:43:50 PST
  Last completed: 2007-01-29 10:43:50 PST
IPCP
  State: Opened
  Last started: 2007-01-29 10:43:50 PST
  Last completed: 2007-01-29 10:43:50 PST
  Negotiated options:
    Local address: 10.10.10.1, Remote address: 10.10.10.2
  Negotiation mode: Active
IPV6CP
  State: Opened
  Last started: 2007-01-29 10:43:50 PST
  Last completed: 2007-01-29 10:43:50 PST
  Negotiated options:
    Local interface identifier: 2a0:a522:64:d319, Remote interface identifier: 0:0:0:c
  Negotiation mode: Passive
```

show ppp interface terse

```
user@host> show ppp interface si-1/3/0 terse
Session name     Session type   Session phase      Session flags
si-1/3/0.0       PPP            Authenticate       Monitored
```

show pppoe interfaces

Syntax
show pppoe interfaces
<brief | detail | extensive>
<pp0.*logical*>

Release Information Command introduced before Junos OS Release 7.4.

Description (J Series Services Routers, M120 routers, M320 routers, and MX Series routers only) Display session-specific information about PPPoE interfaces.

Options
none—Display interface information for all PPPoE interfaces.

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

extensive—(J Series Services Routers) (Optional) Display information about the number of packets sent and received and the number of timeouts during a PPPoE session.

pp0.*logical*—(Optional) Name of an interface. The logical unit number for static interfaces can be a value from 0 through 16385. The logical unit number for dynamic interfaces can be a value from 1073741824 through the maximum number of logical interfaces supported on your router.

Required Privilege Level view

Related Documentation
- Verifying and Managing Agent Circuit Identifier-Based Dynamic VLAN Configuration on page 38

List of Sample Output
show pppoe interfaces on page 732
show pppoe interfaces (Status for the Specified Interface) on page 732
show pppoe interfaces brief on page 733
show pppoe interfaces detail on page 733
show pppoe interfaces extensive (J Series Services Routers only) on page 733
show pppoe interfaces (PPPoE Subscriber Interface with ACI Interface Set) on page 733

Output Fields Table 22 on page 730 lists the output fields for the **show pppoe interfaces** command. Output fields are listed in the approximate order in which they appear. Not all fields are displayed for PPPoE interfaces on M120 and M320 routers in server mode.

Table 22: show pppoe interfaces Output Fields

Field Name	Field Description	Level of Output
Logical Interface		
Logical interface	Name of the logical interface.	All levels
Index	Index number of the logical interface, which reflects its initialization sequence.	**detail extensive** none
State	State of the logical interface: **up** or **down**.	All levels

Table 22: show pppoe interfaces Output Fields *(continued)*

Field Name	Field Description	Level of Output
Session ID	Session ID.	All levels
Type	Origin of the logical interface: **Static** or **Dynamic**. Indicates whether the interface was statically or dynamically created.	**detail extensive** none
Service name	Type of service required (can be used to indicate an ISP name or a class or quality of service).	**detail extensive** none
Configured AC name	Configured access concentrator name.	**detail extensive** none
Session AC name	Name of the access concentrator.	**detail extensive** none
Remote MAC address or **Remote MAC**	MAC address of the remote side of the connection, either the access concentrator or the PPPoE client.	All levels
Auto-reconnect timeout	(J Series Services Routers only) Time after which to try to reconnect after a PPPoE session is terminated, in seconds.	**detail extensive** none
Idle timeout	(J Series Services Routers only) Length of time (in seconds) that a connection can be idle before disconnecting.	**detail extensive** none
Session uptime	Length of time the session has been up, in *hh:mm:ss*.	**detail extensive** none
Dynamic Profile	Name of the dynamic profile that was used to create this interface. If the interface was statically created, this field is not displayed.	**detail extensive** none
Underlying interface	Interface on which PPPoE is running.	All levels
Agent Circuit ID	Agent circuit identifier (ACI) that corresponds to the DSLAM interface that initiated the client service request. An asterisk is interpreted as a wildcard character and can appear at the beginning, the end, or both the beginning and end of the string. If the agent circuit ID is not configured, this field is not displayed.	**detail extensive** none
Agent Remote ID	Agent remote identifier that corresponds to the subscriber associated with the DSLAM interface that initiated the service request. An asterisk is interpreted as a wildcard character and can appear at the beginning, the end, or both at the beginning and end of the string. If the agent remote ID is not configured, this field is not displayed.	**detail extensive** none
ACI Interface Set	Internally-generated name of the dynamic ACI interface set, if configured, and the set index number of the ACI entry in the session database.	**detail extensive** none

Table 22: show pppoe interfaces Output Fields *(continued)*

Field Name	Field Description	Level of Output
Packet Type	Number of packets sent and received during the PPPoE session, categorized by packet type and packet errors: • **PADI**—PPPoE Active Discovery Initiation packets. • **PADO**—PPPoE Active Discovery Offer packets. • **PADR**—PPPoE Active Discovery Request packets. • **PADS**—PPPoE Active Discovery Session-Confirmation packets. • **PADT**—PPPoE Active Discovery Termination packets. • **Service name error**—Packets for which the Service-Name request could not be honored. • **AC system error**—Packets for which the access concentrator experienced an error in performing the host request. For example, the host had insufficient resources to create a virtual circuit. • **Generic error**—Packets that indicate an unrecoverable error occurred. • **Malformed packets**—Malformed or short packets that caused the packet handler to discard the frame as unreadable. • **Unknown packets**—Unrecognized packets.	**extensive**
Timeout	(J Series Services Routers only) Information about timeouts that occurred during the PPPoE session: • **PADI**—No PADO packet has been received within the timeout period. • **PADO**—No PADR packet has been received within the timeout period. (This value is always zero and is not supported.) • **PADR**—No PADS packet has been received within the timeout period.	**extensive**

Sample Output

show pppoe interfaces

```
user@host> show pppoe interfaces
pp0.0 Index 66
  State: Down, Session ID: None,
  Service name: None, Configured AC name: sapphire,
  Session AC name: None, Remote MAC address: 00:00:00:00:00:00,
  Auto-reconnect timeout: 100 seconds, Idle timeout: Never,
  Underlying interface: at-5/0/0.0 Index 71
```

show pppoe interfaces (Status for the Specified Interface)

```
user@host> show pppoe interfaces pp0.1073741827
pp0.1073741827 Index 70
  State: Session Up, Session ID: 30, Type: Dynamic,
  Session AC name: velorum,
  Remote MAC address: 00:90:1A:42:0A:C1,
  Session uptime: 16:45:46 ago,
  Underlying interface: ge-2/0/3.1 Index 73
  Service name: premium
  Dynamic Profile: PppoeProfile
  Agent Circuit ID: velorum-ge-2/0/3
  Agent Remote ID: westford
```


show pppoe interfaces brief

```
user@host> show pppoe interfaces brief
Interface        Underlying           State       Session    Remote
                 interface                        ID         MAC
pp0.0            ge-2/0/3.2           Session Up  27         00:90:1A:42:0A:C1
pp0.1            ge-2/0/3.2           Session Up  28         00:90:1A:42:0A:C1
pp0.1073741824   ge-2/0/3.1           Session Up  29         00:90:1A:42:0A:C1
pp0.1073741825   ge-2/0/3.1           Session Up  30         00:90:1A:42:0A:C1
pp0.1073741826   ge-2/0/3.1           Session Up  31         00:90:1A:42:0A:C1
```

show pppoe interfaces detail

```
user@host> show pppoe interfaces detail
pp0.0 Index 66
  State: Down, Session ID: None, Type: Static,
  Service name: None, Configured AC name: sapphire,
  Session AC name: None, Remote MAC address: 00:00:00:00:00:00,
  Auto-reconnect timeout: 100 seconds, Idle timeout: Never,
  Underlying interface: at-5/0/0.0 Index 71
```

show pppoe interfaces extensive (J Series Services Routers only)

```
user@host> show pppoe interfaces pp0.1 extensive
pp0.1 Index 66
  State: Down, Session ID: 26, Type: Static,
  Service name: None, Configured AC name: sapphire,
  Session AC name: None, Remote MAC address: 00:00:00:00:00:00,
  Auto-reconnect timeout: 100 seconds, Idle timeout: Never,
  Underlying interface: ge-3/0/3.1 Index 71
  PacketType                       Sent         Received
    PADI                              0                0
    PADO                              0                0
    PADR                              0                6
    PADS                              6                0
    PADT                              6                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                              0
    PADO                              0
    PADR                              0
```

show pppoe interfaces (PPPoE Subscriber Interface with ACI Interface Set)

```
user@host> show pppoe interfaces pp0.1073741827
pp0.1073741827 Index 346
  State: Session Up, Session ID: 4, Type: Dynamic,
  Service name: AGILENT, Remote MAC address: 00:00:64:39:01:02,
  Session AC name: nbc,
  Session uptime: 6d 02:22 ago,
  Dynamic Profile: aci-vlan-pppoe-profile,
  Underlying interface: demux0.1073741826 Index 345
  Agent Circuit ID: aci-ppp-dhcp-dvlan-50
  ACI Interface Set: aci-1002-demux0.1073741826 Index 2
```


show pppoe lockout

Syntax

show pppoe lockout
<*underlying-interface-name*>

Release Information

Command introduced in Junos OS Release 11.4.

Description

Display summary information about PPPoE clients currently undergoing lockout or currently in a lockout grace period on all PPPoE underlying logical interfaces or on a specified PPPoE underlying logical interface.

Options

none—Display information about the lockout condition and the lockout grace period for PPPoE clients on all PPPoE underlying logical interfaces.

underlying-interface-name—(Optional) Name of the PPPoE underlying logical interface.

Required Privilege Level

view

Related Documentation

- Verifying and Managing Dynamic PPPoE Configuration on page 211
- Configuring Lockout of PPPoE Subscriber Sessions on page 198

List of Sample Output

show pppoe lockout underlying-interface-name on page 735

Output Fields

Table 23 on page 734 lists the output fields for the **show pppoe lockout** command. Output fields are listed in the approximate order in which they appear.

Table 23: show pppoe lockout Output Fields

Field Name	Field Description
underlying-Interface-name	Name of the PPPoE underlying logical interface.
Index	Index number of the logical interface, which reflects its initialization sequence.
Short-cycle Protection level	State of PPPoE short cycle protection: **On** or **Off**. Enabling short cycle protection, also known as PPPoE lockout, on the PPPoE underlying interface temporarily prevents (locks out) a failed or short-lived (short-cycle) PPPoE subscriber session from reconnecting to the router for a default or configurable period of time. PPPoE client sessions are identified by their unique media access control (MAC) source address.
Lockout Time (seconds)	Displays the PPPoE lockout time range, the number of PPPoE clients in lockout condition, and the number of PPPoE clients in a lockout grace period: • **Min**—Minimum lockout time, in seconds, configured on the PPPoE underlying interface. • **Max**—Maximum lockout time, in seconds, configured on the PPPoE underlying interface. • **Total clients in lockout**—Number of PPPoE clients currently undergoing lockout. • **Total clients in lockout grace period**—Number of PPPoE clients currently in a lockout grace period. A *lockout grace period* occurs when the time between lockout events is greater than either 15 minutes or the maximum lockout time.

Table 23: show pppoe lockout Output Fields *(continued)*

Field Name	Field Description
Client Address	MAC source address of the PPPoE client.
Current	Current lockout time, in seconds; displays **0** (zero) if the PPPoE client is not undergoing lockout.
Elapsed	Time elapsed into the lockout period, in seconds; displays **0** (zero) if the PPPoE client is not undergoing lockout
Next	Lockout time, in seconds, that the router uses for the next lockout event; displays a nonzero value if the PPPoE client is currently in a lockout grace period.

Sample Output

show pppoe lockout underlying-interface-name

```
user@host> show pppoe lockout xe-1/0/0.0
xe-1/0/0.0 Index 10305
  Lockout Time (seconds): Min: 1, Max: 300
    Total clients in lockout: 2
    Total clients in lockout grace period: 1

  Client Address             Current   Elapsed    Next
    02:01:00:00:00:05             16        10      32
    04:01:00:00:00:ab            256       168     300
    0b:cd:ef:00:00:23              0         0       8
```

show pppoe service-name-tables

Syntax

show pppoe service-name-tables
<*table-name*>

Release Information

Command introduced in Junos OS Release 10.0.

Description

(M120 routers, M320 routers, and MX Series routers only) Display configuration information about PPPoE service name tables.

Options

none—Display the names of configured PPPoE service name tables.

table-name—(Optional) Name of a configured PPPoE service name table.

Required Privilege Level

view

Related Documentation

- *Verifying a PPPoE Configuration*
- Verifying and Managing Dynamic PPPoE Configuration on page 211

List of Sample Output

show pppoe service-name-tables on page 737
show pppoe service-name-tables (For the Specified Table Name) on page 737

Output Fields

Table 24 on page 736 lists the output fields for the **show pppoe service-name-tables** command. Output fields are listed in the approximate order in which they appear.

Table 24: show pppoe service-name-tables Output Fields

Field Name	Field Description	Level of Output
Service Name Table	Name of the PPPoE service name table.	none
Service Name	Name of a configured service in the PPPoE service name table: • **<empty>**—Service of zero length that represents an unspecified service • **<any>**—Default service for non-empty service entries that do not match the configured empty or named service entries • ***service-name***—Named service entry	none
Action	Action taken when the PPPoE underlying interface interface receives a PPPoE Active Discovery Initiation (PADI) packet with the specified named service, **empty** service, **any** service, or ACI/ARI pair: • **Delay** ***seconds***—Number of seconds that the interface delays before responding with a PPPoE Active Discovery Offer (PADO) packet • **Drop**—Interface drops (ignores) the packet. • **Terminate**—Interface responds immediately with a PADO packet	none
Dynamic Profile	Name of the dynamic profile with which the router creates a dynamic PPPoE subscriber interface. A dynamic profile can be assigned to a named service, **empty** service, **any** service, or ACI/ARI pair.	none

Table 24: show pppoe service-name-tables Output Fields *(continued)*

Field Name	Field Description	Level of Output
Routing Instance	Name of the routing instance in which to instantiate the dynamic PPPoE subscriber interface. A routing instance can be assigned to a named service, **empty** service, **any** service, or ACI/ARI pair.	none
Max Sessions	Maximum number of active PPPoE sessions that the router can establish with the specified named service, **empty** service, or **any** service.	none
Active Sessions	Current count of active PPPoE sessions created using the specified named service, **empty** service, or **any** service. The Active Sessions value cannot exceed the Max Sessions value.	none
ACI	Agent circuit identifier (ACI) that corresponds to the DSLAM interface that initiated the client service request. An asterisk is interpreted as a wildcard character and can appear at the beginning, the end, or both the beginning and end of the string. An ACI can be configured as part of an ACI/ARI pair for a named service, **empty** service, or **any** service.	none
ARI	Agent remote identifier (ARI) that corresponds to the subscriber associated with the DSLAM interface that initiated the service request. An asterisk is interpreted as a wildcard character and can appear at the beginning, the end, or both at the beginning and end of the string. An ARI can be configured as part of an ACI/ARI pair for a named service, **empty** service, or **any** service.	none
Static Interface	Name of the static PPPoE interface reserved for exclusive use by the PPPoE client with matching ACI/ARI information. A static interface can be configured only for an ACI/ARI pair.	none

Sample Output

show pppoe service-name-tables

```
user@host> show pppoe service-name-tables
Service Name Table: test1
Service Name Table: test2
Service Name Table: test3
```

show pppoe service-name-tables (For the Specified Table Name)

```
user@host> show pppoe service-name-tables Table1
Service Name Table: Table1
  Service Name: <empty>
    Action: Terminate
    Dynamic Profile: BasicPppoeProfile
    Max Sessions: 100
    Active Sessions: 3
  Service Name: <any>
    Action: Drop
    ACI: velorum-ge-2/0/3
    ARI: westford
      Action: Terminate
      Static Interface: pp0.100
    ACI: volantis-ge-5/0/5
    ARI: sunnyvale
```

```
        Action: Terminate
        Static Interface: pp0.101
   Service Name: Wholesale
     Action: Terminate
     Dynamic Profile: WholesalePppoeProfile
     Routing Instance: WholesaleRI
     Max Sessions: 16000
     Active Sessions: 4
```


show pppoe sessions

Syntax

show pppoe sessions
<aci *circuit-id-string*>
<ari *remote-id-string*>
<service *service-name*>

Release Information

Command introduced in Junos OS Release 10.2.

Description

(M120 routers, M320 routers, and MX Series routers only) Display information about all active PPPoE sessions on the router, or about the active PPPoE sessions established for a specified service name, agent circuit identifier (ACI), or agent remote identifier (ARI).

Options

none—Display information for all active PPPoE sessions on the router.

aci ***circuit-id-string***—(Optional) Display information only for active PPPoE sessions established with the specified agent circuit identifier. The agent circuit identifier corresponds to the DSLAM interface that initiated the service request.

ari ***remote-id-string***—(Optional) Display information only for active PPPoE sessions established with the specified agent remote identifier. The agent remote identifier corresponds to the subscriber associated with the DSLAM interface that initiated the service request.

service ***service-name***—(Optional) Display information only for active PPPoE sessions established with the specified service, where *service-name* can be **empty**, **any**, or a named service.

Required Privilege Level

view

Related Documentation

- *Verifying a PPPoE Configuration*
- Verifying and Managing Dynamic PPPoE Configuration on page 211

List of Sample Output

show pppoe sessions (For All Active Sessions) on page 740
show pppoe sessions (For All Active Sessions Matching the Agent Circuit Identifier) on page 740

Output Fields

Table 25 on page 739 lists the output fields for the **show pppoe sessions** command. Output fields are listed in the approximate order in which they appear.

Table 25: show pppoe sessions Output Fields

Field Name	Field Description	Level of Output
Interface	Name of the statically-created or dynamically-created PPPoE interface for the active PPPoE session.	none
Underlying interface	Interface on which PPPoE is running.	none

Table 25: show pppoe sessions Output Fields *(continued)*

Field Name	Field Description	Level of Output
State	State of the PPPoE session; displays **Session Up** for active PPPoE sessions.	none
Session ID	PPPoE session identifier.	none
Remote MAC	MAC address of the remote side of the connection, either the access concentrator or the PPPoE client.	none

Sample Output

show pppoe sessions (For All Active Sessions)

```
user@host> show pppoe sessions
Interface         Underlying             State        Session     Remote
                  interface                           ID          MAC
pp0.0             ge-2/0/3.2             Session Up   27          00:90:1A:42:0A:C1
pp0.1             ge-2/0/3.2             Session Up   28          00:90:1A:42:0A:C1
pp0.1073741824    ge-2/0/3.1             Session Up   29          00:90:1A:42:0A:C1
pp0.1073741825    ge-2/0/3.1             Session Up   30          00:90:1A:42:0A:C1
pp0.1073741826    ge-2/0/3.1             Session Up   31          00:90:1A:42:0A:C1
```

show pppoe sessions (For All Active Sessions Matching the Agent Circuit Identifier)

```
user@host> show pppoe sessions aci "velorum-ge-2/0/3"
Interface         Underlying             State        Session     Remote
                  interface                           ID          MAC
pp0.0             ge-2/0/3.2             Session Up   27          00:90:1A:42:0A:C1
pp0.1             ge-2/0/3.2             Session Up   28          00:90:1A:42:0A:C1
```


show pppoe statistics

Syntax

show pppoe statistics
<*logical-interface-name*>

Release Information

Command introduced before Junos OS Release 7.4.
logical-interface-name option introduced in Junos OS Release 10.1.

Description

(J Series Services Routers, M120 routers, M320 routers, and MX Series routers only) Display statistics information about PPPoE interfaces.

Options

none—Display PPPoE statistics for all interfaces.

logical-interface-name—(Optional) Name of a PPPoE underlying logical interface. Supported for M120 routers, M320 routers, and MX Series routers only.

Required Privilege Level

view

Related Documentation

- *show ppp address-pool*
- show pppoe underlying-interfaces on page 743

List of Sample Output

show pppoe statistics on page 742
show pppoe statistics (For the Specified Underlying Interface Only) on page 742

Output Fields

Table 26 on page 741 lists the output fields for the **show pppoe statistics** command. Output fields are listed in the approximate order in which they appear.

Table 26: show pppoe statistics Output Fields

Field Name	Field Description
Active PPPoE sessions	Total number of active PPPoE sessions and the number of packets sent and received during the PPPoE session, categorized by packet type and packet errors: • **PADI**—PPPoE Active Discovery Initiation packets. • **PADO**—PPPoE Active Discovery Offer packets. • **PADR**—PPPoE Active Discovery Request packets. • **PADS**—PPPoE Active Discovery Session-Confirmation packets. • **PADT**—PPPoE Active Discovery Termination packets. • **Service name error**—Packets for which the Service-Name request could not be honored. • **AC system error**—Packets for which the access concentrator experienced an error in performing the host request. For example, the host had insufficient resources to create a virtual circuit. • **Generic error**—Packets that indicate an unrecoverable error occurred. • **Malformed packets**—Malformed or short packets that caused the packet handler to discard the frame as unreadable. • **Unknown packets**—Unrecognized packets.

Table 26: show pppoe statistics Output Fields *(continued)*

Field Name	Field Description
Timeouts	Information about timeouts that occurred during the PPPoE session (not displayed for M120, M320, and MX Series routers): • **PADI**—No PADR packet has been received within the timeout period. (This value is always zero and is not supported.) • **PADO**—No PPPoE Active Discovery Offer packet has been received within the timeout period. • **PADR**—No PADS packet has been received within the timeout period.

Sample Output

show pppoe statistics

```
user@host> show pppoe statistics
Active PPPoE sessions: 1
  PacketType          Sent      Received
  PADI                   0             0
  PADO                   0             0
  PADR                   0             0
  PADS                   0             0
  PADT                   0             0
  Service name error     0             0
  AC system error        0             0
  Generic error          0             0
  Malformed packets      0             0
  Unknown packets        0             0
  Timeouts
  PADI                   0
  PADO                   0
  PADR                   0
```

show pppoe statistics (For the Specified Underlying Interface Only)

```
user@host> show pppoe statistics ge-4/0/3.2
Active PPPoE sessions: 4
  PacketType                    Sent        Received
    PADI                           0               5
    PADO                           5               0
    PADR                           0               5
    PADS                           4               0
    PADT                           0               1
    Service name error             0               0
    AC system error                0               0
    Generic error                  0               0
    Malformed packets              0               0
    Unknown packets                0               0
```


show pppoe underlying-interfaces

Syntax

show pppoe underlying-interfaces
<brief | detail | extensive>
<lockout>
<*logical-interface-name*>

Release Information

Command introduced in Junos OS Release 10.0.
lockout option added in Junos OS Release 11.4.

Description

(M120, M320, and MX Series routers only) Display information about PPPoE underlying interfaces.

Options

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

lockout—(Optional) Display summary information about the lockout condition and the lockout grace period for PPPoE clients on the PPPoE underlying interface.

logical-interface-name—(Optional) Name of a PPPoE underlying logical interface.

Required Privilege Level

view

Related Documentation

- Verifying and Managing Dynamic PPPoE Configuration on page 211
- Configuring an Underlying Interface for Dynamic PPPoE Subscriber Interfaces on page 156
- Configuring the PPPoE Family for an Underlying Interface on page 159
- Verifying and Managing Agent Circuit Identifier-Based Dynamic VLAN Configuration on page 38

List of Sample Output

show pppoe underlying-interfaces brief on page 746
show pppoe underlying-interfaces detail on page 746
show pppoe underlying-interfaces extensive on page 746
show pppoe underlying-interfaces extensive (PPPoE client in lockout condition) on page 747
show pppoe underlying-interfaces lockout on page 747
show pppoe underlying-interfaces detail (Autosensing Configured for ACI-based Dynamic VLANs) on page 748

Output Fields

Table 27 on page 743 lists the output fields for the **show pppoe underlying-interfaces** command. Output fields are listed in the approximate order in which they appear.

Table 27: show pppoe underlying-interfaces Output Fields

Field Name	Field Description	Level of Output
Underlying Interface	Name of the PPPoE underlying logical interface.	All levels

Table 27: show pppoe underlying-interfaces Output Fields *(continued)*

Field Name	Field Description	Level of Output
Service Name Table	Name of the service name table.	All levels
Dynamic Profile	Name of the dynamic profile that was used to create this interface. If the interface was statically created, then the value is **none**.	All levels
Index	Index number of the logical interface, which reflects its initialization sequence.	**detail extensive**
State	Origin of the logical interface: **Static** or **Dynamic**. Indicates whether the interface was statically or dynamically created.	**detail extensive**
Operational States	Fields in this block are actual operational values rather than simply the configured values. The operational values can be the result of RADIUS-initiated changes.	**detail extensive**
Max Sessions	Maximum number of PPPoE logical interfaces that can be activated on the underlying interface. When this number of logical interfaces has been established, all subsequent PPPoE Active Discovery Initiation (PADI) packets are dropped and all subsequent PPPoE Active Discovery Request (PADR) packets trigger PPPoE Active Discovery Session (PADS) error responses.	**detail extensive**
Max Sessions VSA Ignore	Whether the router is configured to ignore (clear) the PPPoE maximum session value returned by RADIUS in the Max-Clients-Per-Interface Juniper Networks VSA [26-143] and restore the PPPoE maximum session value on the underlying interface to the value configure with the **max-sessions** statement: **Off** (default) or **On**.	**detail extensive** none
Active Sessions	Number of active PPPoE sessions on the underlying interface. If a dynamic profile is listed, then it is the number of active PPPoE sessions on the underlying interface that are using this profile. The Active Sessions value must not exceed the Max Sessions value.	**detail extensive**
Agent Circuit Identifier	Whether the underlying interface is configured to enable creation of (autosense) dynamic VLAN subscriber interfaces based on agent circuit identifier (ACI) information. **Autosensing** indicates that creation of ACI-based dynamic VLAN interfaces is enabled on the underlying interface. If creation of ACI-based dynamic VLANs is not configured on the underlying interface, this field does not appear.	**detail extensive** none
Duplicate Protection	State of PPPoE duplicate protection: **On** or **Off**. When duplicate protection is configured for the underlying interface, a dynamic PPPoE logical interface cannot be activated when an existing active logical interface is present for the same PPPoE client. The uniqueness of the PPPoE client is determined by the client's MAC address.	**detail extensive**
Short Cycle Protection	State of PPPoE short cycle protection: **mac-address**, **circuit-id**, or **Off**. Enabling short cycle protection, also known as PPPoE lockout, on the PPPoE underlying interface temporarily prevents (locks out) a failed or short-lived (short-cycle) PPPoE subscriber session from reconnecting to the router for a default or configurable period of time. PPPoE client sessions are identified by their unique media access control (MAC) source address.	**detail extensive**

Table 27: show pppoe underlying-interfaces Output Fields *(continued)*

Field Name	Field Description	Level of Output
Direct Connect	State of the configuration to ignore DSL Forum VSAs: **On** or **Off**. When configured, the router ignores any of these VSAs received from a directly connected CPE device on the interface.	**detail extensive** none
AC Name	Name of the access concentrator.	**detail extensive**
PacketType	Number of packets sent and received during the PPPoE session, categorized by packet type and packet errors: • **PADI**—PPPoE Active Discovery Initiation packets. • **PADO**—PPPoE Active Discovery Offer packets. • **PADR**—PPPoE Active Discovery Request packets. • **PADS**—PPPoE Active Discovery Session-Confirmation packets. • **PADT**—PPPoE Active Discovery Termination packets. • **Service name error**—Packets for which the Service-Name request could not be honored. • **AC system error**—Packets for which the access concentrator experienced an error in performing the host request. For example, the host had insufficient resources to create a virtual circuit. • **Generic error**—Packets that indicate an unrecoverable error occurred. • **Malformed packets**—Malformed or short packets that caused the packet handler to discard the frame as unreadable. • **Unknown packets**—Unrecognized packets.	**extensive**
Lockout Time (sec)	The PPPoE lockout time range, the number of PPPoE clients in lockout condition, and the number of PPPoE clients in a lockout grace period if **Short Cycle Protection** is enabled (**On**): • **Min**—Minimum lockout time, in seconds, configured on the PPPoE underlying interface. • **Max**—Maximum lockout time, in seconds, configured on the PPPoE underlying interface. • **Total clients in lockout**—Number of PPPoE clients currently undergoing lockout. • **Total clients in lockout grace period**—Number of PPPoE clients currently in a lockout grace period. A *lockout grace period* occurs when the time between lockout events is greater than either 15 minutes or the maximum lockout time.	**extensive**
Client Address	MAC source address of the PPPoE client.	**extensive**
Current	Current lockout time, in seconds; displays **0** (zero) if the PPPoE client is not undergoing lockout.	**extensive**
Elapsed	Time elapsed into the lockout period, in seconds; displays **0**zero) if the PPPoE client is not undergoing lockout	**extensive**
Next	Lockout time, in seconds, that the router uses for the next lockout event; displays a nonzero value if the PPPoE client is currently in a lockout grace period.	**extensive**

Sample Output

show pppoe underlying-interfaces brief

```
user@host> show pppoe underlying-interfaces brief
Underlying Interface     Service Name Table    Dynamic Profile
ge-4/0/3.1               Premium               None
ge-4/0/3.2               None                  PppoeProfile
```

show pppoe underlying-interfaces detail

```
user@host> show pppoe underlying-interfaces detail
ge-4/0/3.1 Index 73
  Operational States:
  State: Static, Dynamic Profile: None,
  Max Sessions: 4000, Max Sessions VSA Ignore: Off,
  Active Sessions: 0,
  Service Name Table: Premium,
  Direct Connect: Off,
  AC Name: velorum, Duplicate Protection: On,
  Short Cycle Protection: Off

ge-4/0/3.2 Index 78
  Operational States:
  State: Dynamic, Dynamic Profile: PppoeProfile,
  Max Sessions: 500, Max Sessions VSA Ignore: Off,
  Active Sessions: 3,
  Service Name Table: None,
  Direct Connect: Off,
  AC Name: velorum, Duplicate Protection: On,
  Short Cycle Protection: Off
```

show pppoe underlying-interfaces extensive

```
user@host> show pppoe underlying-interfaces extensive
ge-4/0/3.1 Index 73
  Operational States:
  State: Static, Dynamic Profile: None,
  Max Sessions: 4000, Max Sessions VSA Ignore Off,
  Active Sessions: 0,
  Service Name Table: None,
  Direct Connect: Off,
  AC Name: velorum, Duplicate Protection: Off,
  Short Cycle Protection: Off

  PacketType                       Sent         Received

    PADI                              0                0
    PADO                              0                0
    PADR                              0                0
    PADS                              0                0
    PADT                              0                0
    Service name error                0                0
    AC system error                   0                0
    Generic error                     0                0
    Malformed packets                 0                0
    Unknown packets                   0                0

ge-4/0/3.2 Index 78
  Operational States:
```

```
  State: Dynamic, Dynamic Profile: PppoeProfile,
  Max Sessions: 4000, Max Sessions VSA Ignore: Off
  Active Sessions: 3,
  Service Name Table: None,
  Direct Connect: Off,
  AC Name: velorum, Duplicate Protection: Off,
  Short Cycle Protection: Off

  PacketType                        Sent          Received
    PADI                               0                 5
    PADO                               5                 0
    PADR                               0                 5
    PADS                               4                 0
    PADT                               0                 1
    Service name error                 0                 0
    AC system error                    0                 0
    Generic error                      0                 0
    Malformed packets                  0                 0
    Unknown packets                    0                 0
```

show pppoe underlying-interfaces extensive (PPPoE client in lockout condition)

```
user@host> show pppoe underlying-interfaces ge-1/0/0/.0 extensive
ge-1/0/0.0 Index 71
  State: Static, Dynamic Profile: None,
  Max Sessions: 32000, Max Sessions VSA Ignore: Off,
  Active Sessions: 0,
  Service Name Table: None,
  Direct Connect: Off,
  AC name: winona, Duplicate Protection: On,
  Short Cycle Protection: Off

  PacketType                        Sent          Received
    PADI                               0                 7
    PADO                               3                 0
    PADR                               0                 3
    PADS                               3                 0
    PADT                               2                 1
    Service name error                 0                 0
    AC system error                    0                 0
    Generic error                      0                 0
    Malformed packets                  0                 0
    Unknown packets                    0                 0

  Lockout Time (sec):  Min: 1, Max: 30
    Total clients in lockout: 1
    Total clients in lockout grace period: 0

  Client Address              Current     Elapsed     Next
    00:10:94:00:00:01               4           3        8
```

show pppoe underlying-interfaces lockout

```
user@host> show pppoe underlying-interfaces ge-1/0/0.0 lockout
ge-1/0/0.0 Index 71
  Short Cycle Protection: Off,
  Lockout Time (sec):  Min: 10, Max: 60
    Total clients in lockout: 0
    Total clients in lockout grace period: 0
```


show pppoe underlying-interfaces detail (Autosensing Configured for ACI-based Dynamic VLANs)

```
user@host> show pppoe underlying-interfaces demux0.1073741826 detail
demux0.1073741826 Index 345
  State: Dynamic, Dynamic Profile: aci-vlan-pppoe-profile,
  Max Sessions: 32000, Max Sessions VSA Ignore: Off,
  Active Sessions: 1,
  Agent Circuit Identifier: Autosensing,
  Service Name Table: None,
  Duplicate Protection: On, Short Cycle Protection: Off,
  Direct Connect: Off,
  AC Name: nbc,
  Short Cycle Protection Level: circuit-id,
```


show services l2tp session

Syntax

show services l2tp session
<brief | detail | extensive>
<interface *interface-name*>
<local-gateway *gateway-address*>
<local-gateway-name *gateway-name*>
<local-session-id *session-id*>
<local-tunnel-id *tunnel-id*>
<peer-gateway *gateway-address*>
<peer-gateway-name *gateway-name*>
<statistics>
<tunnel-group *group-name*>
<user *username*>

Release Information

Command introduced before Junos OS Release 7.4.
Support for LAC on MX Series routers introduced in Junos OS Release 10.4.
Support for LNS on MX Series routers introduced in Junos OS Release 11.4.

Description

(M10i and M7i routers only) Display information about active L2TP sessions for LNS.

(MX Series routers only) Display information about active L2TP sessions for LAC and LNS.

Options

none—Display standard information about all active L2TP sessions.

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

interface *interface-name*—(Optional) Display L2TP session information for only the specified adaptive services or inline services interface. The interface type depends on the line card as follows:

- **si-*fpc/pic/port***— MPCs on MX Series routers only. This option is not available for L2TP on M Series routers.
- **sp-*fpc/pic/port***—AS or Multiservices PICs on M7i, M10i, and M120 routers only. This option is not available for L2TP on MX Series routers.

local-gateway *gateway-address*—(Optional) Display L2TP session information for only the specified local gateway address.

local-gateway-name *gateway-name*—(Optional) Display L2TP session information for only the specified local gateway name.

local-session-id *session-id*—(Optional) Display L2TP session information for only the specified local session identifier.

local-tunnel-id *tunnel-id*—(Optional) Display L2TP session information for only the specified local tunnel identifier.

peer-gateway *gateway-address*—(Optional) Display L2TP session information for only the specified peer gateway address.

peer-gateway-name ***gateway-name***—(Optional) Display L2TP session information for only the specified peer gateway name.

statistics—(Optional) Display the number of control packets and bytes transmitted and received for the session. You cannot include this option with any of the level options, **brief**, **detail**, or **extensive**.

tunnel-group ***group-name***—(Optional) Display L2TP session information for only the specified tunnel group. To display information about L2TP CPU and memory usage, you can include the tunnel group name in the **show services service-sets memory-usage** ***group-name*** and **show services service-sets cpu-usage** ***group-name*** commands. This option is not available for L2TP LAC on MX Series routers.

user ***username***—(M Series routers only) (Optional) Display L2TP session information for only the specified username.

Required Privilege Level

view

Related Documentation

- *L2TP Services Configuration Overview*
- *L2TP Minimum Configuration*
- *clear services l2tp session*

List of Sample Output

show services l2tp session (LNS on M Series Routers) on page 753
show services l2tp session (LNS on MX Series Routers) on page 753
show services l2tp session (LAC) on page 753
show services l2tp session detail (LAC) on page 753
show services l2tp session extensive (LAC) on page 754
show services l2tp session extensive (LAC on MX Series Routers) on page 754
show services l2tp session extensive (LNS on M Series Routers) on page 754
show services l2tp session extensive (LNS on MX Series Routers) on page 755
show services l2tp session statistics (MX Series Routers) on page 755

Output Fields

Table 28 on page 750 lists the output fields for the **show services l2tp session** command. Output fields are listed in the approximate order in which they appear.

Table 28: show services l2tp session Output Fields

Field Name	Field Description	Level of Output
Interface	(LNS only) Name of an adaptive services interface.	All levels
Tunnel group	(LNS only) Name of a tunnel group.	All levels
Tunnel local ID	Identifier of the local endpoint of the tunnel, as assigned by the L2TP network server (LNS).	All levels
Session local ID	Identifier of the local endpoint of the L2TP session, as assigned by the LNS.	All levels

Table 28: show services l2tp session Output Fields *(continued)*

Field Name	Field Description	Level of Output
Session remote ID	Identifier of the remote endpoint of the L2TP session, as assigned by the L2TP access concentrator (LAC).	All levels
State	State of the L2TP session: • **Established**—Session is operating. This is the only state supported for the LAC. • **closed**—Session is being closed. • **destroyed**—Session is being destroyed. • **clean-up**—Session is being cleaned up. • **lns-ic-accept-new**—New session is being accepted. • **lns-ic-idle**—Session has been created and is idle. • **lns-ic-reject-new**—New session is being rejected. • **lns-ic-wait-connect**—Session is waiting for the peer's incoming call connected (ICCN) message.	All levels
Bundle ID	(LNS only) Bundle identifier. Indicates the session is part of a multilink bundle. Sessions that have a blank **Bundle** field are not participating in the Multilink Protocol. Sessions in a multilink bundle might belong to different L2TP tunnels. For L2TP output organized by bundle ID, issue the **show services l2tp multilink extensive** command.	All levels
Mode	(LNS) Mode of the interface representing the session: **shared** or **exclusive**. (LAC) Mode of the interface representing the session: **shared** or **dedicated**. Only **dedicated** is currently supported for the LAC.	**extensive**
Local IP	IP address of local endpoint of the Point-to-Point Protocol (PPP) session.	**extensive**
Remote IP	IP address of remote endpoint of the PPP session.	**extensive**
Username	(LNS only) Name of the user logged in to the session.	All levels
Assigned IP address	(LNS only) IP address assigned to remote client.	**extensive**
Local name	For LNS, name of the LNS instance in which the session was created. For LAC, name of the LAC.	**extensive**
Remote name	For LNS, name of the LAC from which the session was created. For LAC, name of the LAC instance.	**extensive**
Local MRU	(LNS only) Maximum receive unit (MRU) setting of the local device, in bytes.	**extensive**
Remote MRU	(LNS only) MRU setting of the remote device, in bytes.	**extensive**
Tx speed	Transmit speed of the session conveyed from the LAC to the LNS, in bits per second (bps). Both the initial (**initial**) and current (**update**) line speeds can be displayed on MX Series routers.	**extensive**

Table 28: show services l2tp session Output Fields *(continued)*

Field Name	Field Description	Level of Output
Rx speed	Receive speed of the session conveyed from the LAC to the LNS, in bits per second (bps). Both the initial (**initial**) and current (**update**) line speeds can be displayed on MX Series routers.	**extensive**
Bearer type	Type of bearer enabled: • **0**—Might indicate that the call was not received over a physical link (for example, when the LAC and PPP are located in the same subsystem). • **1**—Digital access requested. • **2**—Analog access requested. • **4**—Asynchronous Transfer Mode (ATM) bearer support.	**extensive**
Framing type	Type of framing enabled: • **1**—Synchronous framing • **2**—Asynchronous framing	**extensive**
LCP renegotiation	(LNS only) Whether Link Control Protocol (LCP) renegotiation is configured: **On** or **Off**.	**extensive**
Authentication	Type of authentication algorithm used: Challenge Handshake Authentication Protocol (CHAP) or Password Authentication Protocol (PAP).	**extensive**
Interface ID	(LNS only) Identifier used to look up the logical interface for this session.	**extensive**
Interface unit	Logical interface for this session.	All levels
Call serial number	Unique serial number assigned to the call.	**extensive**
Policer bandwidth	Maximum policer bandwidth configured for this session.	**extensive**
Policer burst size	Maximum policer burst size configured for this session.	**extensive**
Firewall filter	Configured firewall filter name.	**extensive**
Session encapsulation overhead	Overhead allowance configured for this session, in bytes.	**extensive**
Session cell overhead	Cell overhead activation (**On** or **Off**).	**extensive**
Create time	Date and time when the call was created.	**extensive**
Up time	Length of time elapsed since the call became active, in hours, minutes, and seconds.	**extensive**

Table 28: show services l2tp session Output Fields *(continued)*

Field Name	Field Description	Level of Output
Idle time	Length of time elapsed since the call became idle, in hours, minutes, and seconds.	**extensive**
Statistics since	Date and time when collection of the following statistics began: • **Control Tx**—Amount of control information transmitted, in packets and bytes. • **Control Rx**—Amount of control information received, in packets and bytes. • **Data Tx**—Amount of data transmitted, in packets and bytes. • **Data Rx**—Amount of data received, in packets and bytes. • **Errors Tx**—Number of errors transmitted, in packets. • **Errors Rx**—Number of errors received, in packets. • **LCP echo req Tx**—Number of LCP echo requests transmitted, in packets. • **LCP echo req Rx**—Number of LCP echo requests received, in packets. • **LCP echo rep Tx**—Number of LCP echo responses transmitted, in packets. • **LCP echo rep Rx**—Number of LCP echo responses received, in packets. • **LCP echo Req timout**—Number of LCP echo requests that timed out. • **LCP echo Req error**—Number of errors received for LCP echo packets. • **LCP echo Rep error** —Number of errors transmitted for LCP echo packets.	**extensive**

Sample Output

show services l2tp session (LNS on M Series Routers)

```
user@host> show services l2tp session
Interface: sp-1/2/0, Tunnel group: group1, Tunnel local ID:  8802
  Local Remote Interface State               Bundle Username
  ID    ID     unit
  37966      5        2 Established
```

show services l2tp session (LNS on MX Series Routers)

```
user@host> show services l2tp session
Tunnel local ID: 40553
  Local  Remote   State                Interface        Interface
  ID     ID                            unit             Name
  17967  1        Established          1073749824       si-5/2/0
```

show services l2tp session (LAC)

```
user@host> show services l2tp session
Tunnel local ID: 31889
  Local  Remote   State                Interface        Interface
  ID     ID                            unit             Name
   31694      1   Established                311        pp0
```

show services l2tp session detail (LAC)

```
user@host> show services l2tp session detail
Tunnel local ID: 31889
  Session local ID: 31694, Session remote ID:     1, Interface unit: 311
    State: Established, Interface: pp0, Mode: Dedicated
```



```
    Local IP: 10.1.1.2:1701, Remote IP: 10.1.1.1:1701
    Local name: ce-lac, Remote name: ce-lns
```

show services l2tp session extensive (LAC)

```
user@host> show services l2tp session extensive
Tunnel local ID: 31889
  Session local ID: 31694, Session remote ID:     1
    Interface unit: 311
    State: Established, Mode: Dedicated
    Local IP: 10.10.1.2:1701, Remote IP: 10.10.1.1:1701
    Local name: ce-lac, Remote name: ce-lns
    Tx speed: 0, Rx speed: 0
    Bearer type: 1, Framing type: 1
    LCP renegotiation: N/A, Authentication: None, Interface ID: N/A
    Interface unit: 311, Call serial number: 0
    Policer bandwidth: 0, Policer burst size: 0
    Policer exclude bandwidth: 0, Firewall filter: 0
    Session encapsulation overhead: 0, Session cell overhead: 0
    Create time: Tue Aug 24 14:38:23 2010, Up time: 01:06:25
    Idle time: N/A
```

show services l2tp session extensive (LAC on MX Series Routers)

```
user@host> show services l2tp session extensive
Tunnel local ID: 31889
  Session local ID: 31694, Session remote ID:     1
    Interface unit: 311
    State: Established, Mode: Dedicated
    Local IP: 10.10.1.2:1701, Remote IP: 10.10.1.1:1701
    Local name: ce-lac, Remote name: ce-lns
    Tx speed: initial 64000, Update 256000
    Rx speed: initial 64000, Update 256000
    Bearer type: 1, Framing type: 1
    LCP renegotiation: N/A, Authentication: None, Interface ID: N/A
    Interface unit: 311, Call serial number: 0
    Policer bandwidth: 0, Policer burst size: 0
    Policer exclude bandwidth: 0, Firewall filter: 0
    Session encapsulation overhead: 0, Session cell overhead: 0
    Create time: Tue Aug 24 14:38:23 2010, Up time: 01:06:25
    Idle time: N/A
```

show services l2tp session extensive (LNS on M Series Routers)

```
user@host> show services l2tp session extensive
Interface: sp-1/2/0, Tunnel group: group1, Tunnel local ID: 62746
  Session local ID: 56793, Session remote ID: 53304
    State: Established, Bundle ID: 5, Mode: shared
    Local IP: 10.128.1.1:1701, Remote IP: 10.128.1.2:1701
    Username: usr1@juniper_1.net, Assigned IP address: 10.50.2.1/32
    Local MRU: 4000, Remote MRU: 1500, Tx speed: 64000, Rx speed: 64000
    Bearer type: 2, Framing type: 1
    LCP renegotiation: Off, Authentication: CHAP, Interface ID: unit_20
    Interface unit: 20, Call serial number: 4137941434
    Policer bandwidth: 64000, Policer burst size: 51200
    Firewall filter: f1
    Session encapsulation overhead: 16, Session cell overhead: On
    Create time: Tue Mar 23 14:13:15 2004, Up time: 01:16:41
    Idle time: 00:00:00
    Statistics since: Tue Mar 23 14:13:13 2004
                        Packets       Bytes
      Control Tx              4          88
```



```
    Control Rx               2           28
    Data Tx                  0            0
    Data Rx                461        29.0k
    Errors Tx                0
    Errors Rx                0

Interface: sp-1/2/0, Tunnel group: group_company_dns, Tunnel local ID: 37266
  Session local ID: 39962, Session remote ID: 53303
    State: Established, Bundle ID: 5, Mode: shared
    Local IP: 10.128.11.1:1701, Remote IP: 10.128.11.2:1701
    Username: usr1@company.com, Assigned IP address: 10.46.2.3/24
    Local name: router-1, Remote name: router-2
    Local MRU: 4470, Remote MRU: 4470, Tx speed: 155000000, Rx speed: 155000000
    Bearer type: 2, Framing type: 1
    LCP renegotiation: Off, Authentication: CHAP, Interface ID: unit_31
    Interface unit: 31, Call serial number: 4137941433
    Policer bandwidth: 64000, Policer burst size: 51200
    Firewall filter: f1
    Create time: Tue Mar 23 14:13:17 2004, Up time: 01:16:39
    Idle time: 01:16:36
    Statistics since: Tue Mar 23 14:13:15 2004
                        Packets        Bytes
      Control Tx               6          196
      Control Rx               4          150
      Data Tx                  0            0
      Data Rx                  1           80
      Errors Tx                0
      Errors Rx                0
```

show services l2tp session extensive (LNS on MX Series Routers)

```
user@host> show services l2tp session extensive
Tunnel local ID: 40553
  Session local ID: 17967, Session remote ID: 1
    Interface unit: 1073749824
    State: Established
    Interface: si-5/2/0
    Mode: Dedicated
    Local IP: 11.1.1.2:1701, Remote IP: 11.1.1.3:1701
    Local name: lns-mx960, Remote name: testlac
    Tx speed: 56000, Rx speed: 0
    Bearer type: 2, Framing type: 1
    LCP renegotiation: Off, Authentication: None
    Call serial number: 1
    Create time: Mon Apr 25 20:27:50 2011, Up time: 00:01:48
    Idle time: N/A
    Statistics since: Mon Apr 25 20:27:50 2011
                        Packets        Bytes
      Control Tx               4          219
      Control Rx               4          221
      Data Tx                  0            0
      Data Rx                 10          228
      Errors Tx                0
      Errors Rx                0
```

show services l2tp session statistics (MX Series Routers)

```
user@host>show services l2tp session statistics local session-id 1
Tunnel local ID: 17185
Session local ID: 1, Session remote ID: 14444, Interface unit: 1073788352
State: Established
```

```
Statistics since: Mon Aug 1 13:27:47 2011
    Packets   Bytes
Data Tx   4    51
Data Rx   3    36
```


show subscribers

Syntax

show subscribers
<detail | extensive | terse>
<aci-interface-set-name *aci-interface-set-name*>
<address *address*>
<agent-circuit-identifier *agent-circuit-identifier-substring*>
<client-type *client-type*>
<count>
<id>
<interface *interface*>
<logical-system *logical-system*>
<mac-address *mac-address*>
<physical-interface *physical-interface-name*>
<profile-name *profile-name*>
<routing-instance *routing-instance*>
<stacked-vlan-id *stacked-vlan-id*>
<subscriber-state *subscriber-state*>
<user-name *user-name*>
<vci *vci-identifier*>
<vpi *vpi-identifier*>
<vlan-id *vlan-id*>

Release Information

Command introduced in Junos OS Release 9.3.
Command introduced in Junos OS Release 9.3 for EX Series switches.
client-type, **mac-address**, **subscriber-state**, and **extensive** options introduced in Junos OS Release 10.2.
count option usage with other options introduced in Junos OS Release 10.2.
Command introduced in Junos OS Release 11.1 for the QFX Series.
Options **aci-interface-set-name** and **agent-circuit-identifier** introduced in Junos OS Release 12.2.
The **physical-interface** and **user-name** options introduced in Junos OS Release 12.3.
Options **vci** and **vpi** introduced in Junos OS Release 12.3R3 and supported in later 12.3R*x* releases.
Options **vci** and **vpi** supported in Junos OS Release 13.2 and later releases. (Not supported in Junos OS Release 13.1.)
Command introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

Display information for active subscribers.

Options

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

aci-interface-set-name—(Optional) Display all dynamic subscriber sessions that use the specified agent circuit identifier (ACI) interface set. Use the ACI interface set name generated by the router, such as aci-1003-ge-1/0/0.4001, and not the actual ACI value found in the DHCP or PPPoE control packets.

address—(Optional) Display subscribers whose IP address matches the specified address. You must specify the IPv4 or IPv6 address prefix without a netmask (for example, 192.168.17.1). If you specify the IP address as a prefix with a netmask (for example, 192.168.17.1/32), the router displays a message that the IP address is invalid, and rejects the command.

agent-circuit-identifier-substring—(Optional) Display all dynamic subscriber sessions whose ACI value matches the specified substring.

client-type—(Optional) Display subscribers whose client type matches the specified client type (DHCP, L2TP, PPP, PPPOE, VLAN, or static).

count—(Optional) Display the count of total subscribers and active subscribers for any specified option. You can use the **count** option alone or with the **address**, **client-type**, **interface**, **logical-system**, **mac-address**, **profile-name**, **routing-instance**, **stacked-vlan-id**, **subscriber-state**, or **vlan-id** options.

id—(Optional) Display a specific subscriber session whose session id matches the specified subscriber ID. You can display subscriber IDs by using the **show subscribers extensive** or the **show subscribers interface extensive** commands.

interface—(Optional) Display subscribers whose interface matches the specified interface.

logical-system—(Optional) Display subscribers whose logical system matches the specified logical system.

mac-address—(Optional) Display subscribers whose MAC address matches the specified MAC address.

physical-interface-name—(M120, M320, and MX Series routers only) (Optional) Display subscribers whose physical interface matches the specified physical interface.

profile-name—(Optional) Display subscribers whose dynamic profile matches the specified profile name.

routing-instance—(Optional) Display subscribers whose routing instance matches the specified routing instance.

stacked-vlan-id—(Optional) Display subscribers whose stacked VLAN ID matches the specified stacked VLAN ID.

subscriber-state—(Optional) Display subscribers whose subscriber state matches the specified subscriber state (ACTIVE, CONFIGURED, INIT, TERMINATED, or TERMINATING).

user-name—(M120, M320, and MX Series routers only) (Optional) Display subscribers whose username matches the specified subscriber name.

vci-identifier—(MX Series routers with MPCs and ATM MICs with SFP only) (Optional) Display active ATM subscribers whose ATM virtual circuit identifier (VCI) matches the specified VCI identifier. The range of values is **0** through **255**.

vpi-identifier—(MX Series routers with MPCs and ATM MICs with SFP only) (Optional) Display active ATM subscribers whose ATM virtual path identifier (VPI) matches the specified VPI identifier. The range of values is **0** through **65535**.

vlan-id—(Optional) Display subscribers whose VLAN ID matches the specified VLAN ID, regardless of whether the subscriber uses a single-tagged or double-tagged VLAN. For subscribers using a double-tagged VLAN, this option displays subscribers where the inner VLAN tag matches the specified VLAN ID. To display only subscribers where the specified value matches only double-tagged VLANs, use the **stacked-vlan-id** ***stacked-vlan-id*** option to match the outer VLAN tag.

NOTE: Due to display limitations, logical system and routing instance output values are truncated when necessary.

Required Privilege Level view

Related Documentation

- show subscribers summary on page 776
- Verifying and Managing Agent Circuit Identifier-Based Dynamic VLAN Configuration on page 38

List of Sample Output

show subscribers routing-instance inst1 count on page 773
show subscribers stacked-vlan-id detail on page 773
show subscribers stacked-vlan-id vlan-id detail (Combined Output) on page 773
show subscribers stacked-vlan-id vlan-id interface detail (Combined Output for a Specific Interface) on page 773
show subscribers user-name detail on page 774
show subscribers vlan-id on page 774
show subscribers vlan-id detail on page 774
show subscribers vpi vci extensive (PPPoE-over-ATM Subscriber Session) on page 774

Output Fields Table 29 on page 760 lists the output fields for the **show subscribers** command. Output fields are listed in the approximate order in which they appear.

Table 29: show subscribers Output Fields

Field Name	Field Description
Interface	Interface associated with the subscriber. The router or switch displays subscribers whose interface matches or begins with the specified interface. The * character indicates a continuation of addresses for the same session.
IP Address/VLAN ID	Subscriber IP address or VLAN ID associated with the subscriber in the form *tpid.vlan-id* No IP address or VLAN ID is assigned to an L2TP tunnel-switched session. For these subscriber sessions the value is **Tunnel-switched**.
User Name	Name of subscriber.
LS:RI	Logical system and routing instance associated with the subscriber.
Type	Subscriber client type (DHCP, L2TP, PPP, PPPoE, STATIC-INTERFACE, VLAN).
IP Address	Subscriber IPv4 address.
IP Netmask	Subscriber IP netmask.
Primary DNS Address	IP address of primary DNS server.
Secondary DNS Address	IP address of secondary DNS server.
Primary WINS Address	IP address of primary WINS server.
Secondary WINS Address	IP address of secondary WINS server.
IPv6 Address	Subscriber IPv6 address, or multiple addresses.
IPv6 Prefix	Subscriber IPv6 prefix. If you are using DHCPv6 prefix delegation, this is the delegated prefix.
IPv6 User Prefix	IPv6 prefix obtained through ND/RA.

Table 29: show subscribers Output Fields *(continued)*

Field Name	Field Description
IPv6 Address Pool	Subscriber IPv6 address pool. The IPv6 address pool is used to allocate IPv6 prefixes to the DHCPv6 clients.
IPv6 Network Prefix Length	Length of the network portion of the IPv6 address.
IPv6 Prefix Length	Length of the subscriber IPv6 prefix.
Logical System	Logical system associated with the subscriber.
Routing Instance	Routing instance associated with the subscriber.
Interface Type	Whether the subscriber interface is **Static** or **Dynamic**.
Interface Set	Internally generated name of the dynamic ACI interface set used by the subscriber session.
Interface Set Type	Interface type of the ACI interface set: **Dynamic**. This is the only ACI interface set type currently supported.
Interface Set Session ID	Identifier of the dynamic ACI interface set entry in the session database.
Underlying Interface	Name of the underlying interface for the subscriber session.
Dynamic Profile Name	Dynamic profile used for the subscriber.
Dynamic Profile Version	Version number of the dynamic profile used for the subscriber.
MAC Address	MAC address associated with the subscriber.
State	Current state of the subscriber session (**Init**, **Configured**, **Active**, **Terminating**, **Tunneled**).
L2TP State	Current state of the L2TP session, **Tunneled** or **Tunnel-switched**. When the value is **Tunnel-switched**, two entries are displayed for the subscriber; the first entry is at the LNS interface on the LTS and the second entry is at the LAC interface on the LTS.
Tunnel switch Profile Name	Name of the L2TP tunnel switch profile that initiates tunnel switching.
Local IP Address	IP address of the local gateway (LAC).
Remote IP Address	IP address of the remote peer (LNS).
VLAN Id	VLAN ID associated with the subscriber in the form *tpid.vlan-id*.
Stacked VLAN Id	Stacked VLAN ID associated with the subscriber in the form *tpid.vlan-id*.
RADIUS Accounting ID	RADIUS accounting ID associated with the subscriber.

Table 29: show subscribers Output Fields *(continued)*

Field Name	Field Description
Agent Circuit ID	Option 82 agent circuit ID associated with the subscriber. The ID is displayed as an ASCII string unless the value has nonprintable characters, in which case it is displayed in hexadecimal format.
Agent Remote ID	Option 82 agent remote ID associated with the subscriber. The ID is displayed as an ASCII string unless the value has nonprintable characters, in which case it is displayed in hexadecimal format.
DHCP Relay IP Address	IP address used by the DHCP relay agent.
ATM VPI	(MX Series routers with MPCs and ATM MICs with SFP only) ATM virtual path identifier (VPI) on the subscriber's physical interface.
ATM VCI	(MX Series routers with MPCs and ATM MICs with SFP only) ATM virtual circuit identifier (VCI) for each VPI configured on the subscriber interface.
Login Time	Date and time at which the subscriber logged in.
Effective shaping-rate	Actual downstream traffic shaping rate for the subscriber, in kilobits per second.
IPv4 rpf-check Fail Filter Name	Name of the filter applied by the dynamic profile to IPv4 packets that fail the RPF check.
IPv6 rpf-check Fail Filter Name	Name of the filter applied by the dynamic profile to IPv6 packets that fail the RPF check.
DHCP Options	**len** = number of hex values in the message. The hex values specify the type, length, value (TLV) for DHCP options, as defined in RFC 2132.
Session ID	ID number for a subscriber service session.
Underlying Session ID	For DHCPv6 subscribers on a PPPoE network, displays the session ID of the underlying PPPoE interface.
Service Sessions	Number of service sessions (that is, a service activated using RADIUS CoA) associated with the subscribers.
Service Session Name	Service session profile name.
Session Timeout (seconds)	Number of seconds of access provided to the subscriber before the session is automatically terminated.
Idle Timeout (seconds)	Number of seconds subscriber can be idle before the session is automatically terminated.
IPv6 Delegated Address Pool	Name of the pool used for DHCPv6 prefix delegation.
IPv6 Delegated Network Prefix Length	Length of the prefix configured for the IPv6 delegated address pool.
IPv6 Interface Address	Address assigned by the Framed-Ipv6-Prefix AAA attribute.

Table 29: show subscribers Output Fields *(continued)*

Field Name	Field Description
IPv6 Framed Interface Id	Interface ID assigned by the Framed-Interface-Id AAA attribute.
ADF IPv4 Input Filter Name	Name assigned to the Ascend-Data-Filter (ADF) interface IPv4 input filter (client or service session). The filter name is followed by the rules (in hexadecimal format) associated with the ADF filter and the decoded rule in Junos OS filter style.
ADF IPv4 Output Filter Name	Name assigned to the Ascend-Data-Filter (ADF) interface IPv4 output filter (client or service session). The filter name is followed by the rules (in hexadecimal format) associated with the ADF filter and the decoded rule in Junos OS filter style.
ADF IPv6 Input Filter Name	Name assigned to the Ascend-Data-Filter (ADF) interface IPv6 input filter (client or service session). The filter name is followed by the rules (in hexadecimal format) associated with the ADF filter and the decoded rule in Junos OS filter style.
ADF IPv6 Output Filter Name	Name assigned to the Ascend-Data-Filter (ADF) interface IPv6 output filter (client or service session). The filter name is followed by the rules (in hexadecimal format) associated with the ADF filter and the decoded rule in Junos OS filter style.
IPv4 Input Filter Name	Name assigned to the IPv4 input filter (client or service session).
IPv4 Output Filter Name	Name assigned to the IPv4 output filter (client or service session).
IPv6 Input Filter Name	Name assigned to the IPv6 input filter (client or service session).
IPv6 Output Filter Name	Name assigned to the IPv6 output filter (client or service session).
IFL Input Filter Name	Name assigned to the logical interface input filter (client or service session).
IFL Output Filter Name	Name assigned to the logical interface output filter (client or service session).

Sample Output

show subscribers (IPv4)

```
user@host> show subscribers
Interface                IP Address/VLAN ID   User Name         LS:RI
ge-1/3/0.1073741824      100                                    default:default
demux0.1073741824        10.0.0.10           WHOLESALER-CLIENT default:default
demux0.1073741825        11.0.0.3            RETAILER1-CLIENT  test1:retailer1
demux0.1073741826        12.0.0.3            RETAILER2-CLIENT  test1:retailer2
```

show subscribers (IPv6)

```
user@host> show subscribers
Interface          IP Address/VLAN ID   User Name           LS:RI
ge-1/0/0.0         2001:db8::c0:0:0:0/74    WHOLESALER-CLIENT   default:default
     *             2001:db8::1/128          subscriber-25       default:default
```

show subscribers (IPv4 and IPv6 Dual Stack)

```
user@host> show subscribers
Interface           IP Address/VLAN ID         User Name
LS:RI
demux0.1073741834   0x8100.1002 0x8100.1
default:default
demux0.1073741835   0x8100.1001 0x8100.1
default:default
pp0.1073741836      61.1.1.1                 dualstackuser1@EXAMPLE1.com
default:ASP-1
*                   2041:1:1::/48
*                   2061:1:1:1::/64
pp0.1073741837      23.1.1.3                 dualstackuser2@EXAMPLE1.com
default:ASP-1
*                   2001:db8:1:2:5::/64
```

show subscribers (LNS on MX Series Routers)

```
user@host> show subscribers
Interface          IP Address/VLAN ID   User Name          LS:RI
si-4/0/0.1         192.168.4.1          xyz@example.com    default:default
```

show subscribers (L2TP Switched Tunnels)

```
user@host> show subscribers
Interface           IP Address/VLAN ID    User Name              LS:RI
si-2/1/0.1073741842 Tunnel-switched       ap@example.com        default:default

si-2/1/0.1073741843 Tunnel-switched       ap@example.com        default:default
```

show subscribers client-type dhcp detail

```
user@host> show subscribers client-type dhcp detail
Type: DHCP
IP Address: 10.20.9.7
IP Netmask: 255.255.0.0
Logical System: default
Routing Instance: default
Interface: demux0.1073744127
Interface type: Dynamic
Dynamic Profile Name: dhcp-demux-prof
MAC Address: 00:10:95:00:00:98
State: Active
Radius Accounting ID: jnpr :2304
Login Time: 2009-08-25 14:43:52 PDT

Type: DHCP
IP Address: 10.20.10.7
IP Netmask: 255.255.0.0
Logical System: default
Routing Instance: default
Interface: demux0.1073744383
Interface type: Dynamic
Dynamic Profile Name: dhcp-demux-prof
MAC Address: 00:10:94:00:01:f3
State: Active
```



```
Radius Accounting ID: jnpr :2560
Login Time: 2009-08-25 14:43:56 PDT
```

show subscribers count

```
user@host> show subscribers count
Total Subscribers: 188, Active Subscribers: 188
```

show subscribers address detail (IPv6)

```
user@host> show subscribers address 10.16.12.137 detail
Type: PPPoE
User Name: pppoeTerV6User1Svc
IP Address: 10.16.12.137
IP Netmask: 255.0.0.0
IPv6 User Prefix: 1016:0:0:c88::/64
Logical System: default
Routing Instance: default
Interface: pp0.1073745151
Interface type: Dynamic
Underlying Interface: demux0.8201
Dynamic Profile Name: pppoe-client-profile
MAC Address: 00:0d:02:01:00:01
Session Timeout (seconds): 31622400
Idle Timeout (seconds): 86400
State: Active
Radius Accounting ID: jnpr demux0.8201:6544
Session ID: 6544
Agent Circuit ID: ifl3720
Agent Remote ID: ifl3720
Login Time: 2012-05-21 13:37:27 PDT
Service Sessions: 1
```

show subscribers detail (IPv4)

```
user@host> show subscribers detail
Type: DHCP
IP Address: 10.20.9.7
IP Netmask: 255.255.0.0
Primary DNS Address: 192.168.17.1
Secondary DNS Address: 192.168.17.2
Primary WINS Address: 192.168.22.1
Secondary WINS Address: 192.168.22.2
Logical System: default
Routing Instance: default
Interface: demux0.1073744127
Interface type: Dynamic
Dynamic Profile Name: dhcp-demux-prof
MAC Address: 00:10:95:00:00:98
State: Active
Radius Accounting ID: jnpr :2304
Idle Timeout (seconds): 600
Login Time: 2009-08-25 14:43:52 PDT
DHCP Options: len 52
35 01 01 39 02 02 40 3d 07 01 00 10 94 00 00 08 33 04 00 00
00 3c 0c 15 63 6c 69 65 6e 74 5f 50 6f 72 74 20 2f 2f 36 2f
33 2d 37 2d 30 37 05 01 06 0f 21 2c
Service Sessions: 2
```

show subscribers detail (IPv6)

```
user@host> show subscribers detail
Type: DHCP
User Name: pd-user1
IPv6 Prefix: 2001:db8:db2:ffff:1::/64
Logical System: default
Routing Instance: default
Interface: ge-3/1/3.2
Interface type: Static
MAC Address: 00:51:ff:ff:00:03
State: Active
Radius Accounting ID: 1
Session ID: 1
Login Time: 2011-08-25 12:12:26 PDT
DHCP Options: len 42
00 08 00 02 00 00 00 01 00 0a 00 03 00 01 00 51 ff ff 00 03
00 06 00 02 00 19 00 19 00 0c 00 00 00 00 00 00 00 00 00 00
00 00
```

show subscribers detail (IPv6 Static Demux Interface)

```
user@host> show subscribers detail
Type: STATIC-INTERFACE
User Name: demux0.1@example.net
IPv6 Prefix: 1:2:3:4:5:6:7:aa/128
Logical System: default
Routing Instance: default
Interface: demux0.1
Interface type: Static
Dynamic Profile Name: junos-default-profile
State: Active
Radius Accounting ID: 185
Login Time: 2010-05-18 14:33:56 EDT
```

show subscribers detail (L2TP LNS Subscribers on MX Series Routers)

```
user@host> show subscribers detail
Type: L2TP
User Name: user1@example.net
IP Address: 10.1.32.58
IP Netmask: 255.255.0.0
Logical System: default
Routing Instance: default
Interface: si-5/2/0.1073749824
Interface type: Dynamic
Dynamic Profile Name: dyn-lns-profile2
Dynamic Profile Version: 1
State: Active
Radius Accounting ID: 8001
Session ID: 8001
Login Time: 2011-04-25 20:27:50 IST
```

show subscribers detail (L2TP Switched Tunnels)

```
user@host> show subscribers detail
Type: L2TP
User Name: ap@example.com
Logical System: default
Routing Instance: default
Interface: si-2/1/0.1073741842
```

```
Interface type: Dynamic
Dynamic Profile Name: dyn-lts-profile
State: Active
L2TP State: Tunnel-switched
Tunnel switch Profile Name: ce-lts-profile
Local IP Address: 10.50.1.1
Remote IP Address: 192.168.20.3
Radius Accounting ID: 21
Session ID: 21
Login Time: 2013-01-18 03:01:11 PST

Type: L2TP
User Name: ap@example.com
Logical System: default
Routing Instance: default
Interface: si-2/1/0.1073741843
Interface type: Dynamic
Dynamic Profile Name: dyn-lts-profile
State: Active
L2TP State: Tunnel-switched
Tunnel switch Profile Name: ce-lts-profile
Local IP Address: 10.30.1.1
Remote IP Address: 172.20.1.10
Session ID: 22
Login Time: 2013-01-18 03:01:14 PST
```

show subscribers detail (Tunneled Subscriber)

```
user@host> show subscribers detail
Type: PPPoE
User Name: user1@example.com
Logical System: default
Routing Instance: default
Interface: pp0.1
State: Active, Tunneled
Radius Accounting ID: 512
```

show subscribers detail (IPv4 and IPv6 Dual Stack)

```
user@host> show subscribers detail
Type: VLAN
Logical System: default
Routing Instance: default
Interface: demux0.1073741824
Interface type: Dynamic
Dynamic Profile Name: svlanProfile
State: Active
Session ID: 1
Stacked VLAN Id: 0x8100.1001
VLAN Id: 0x8100.1
Login Time: 2011-11-30 00:18:04 PST

Type: PPPoE
User Name: dualstackuser1@EXAMPLE1.com
IP Address: 61.1.1.1
IPv6 Prefix: 2041:1:1::/48
IPv6 User Prefix: 2061:1:1:1::/64
Logical System: default
Routing Instance: ASP-1
Interface: pp0.1073741825
Interface type: Dynamic
```



```
Dynamic Profile Name: dualStack-Profile1
MAC Address: 00:00:64:03:01:02
State: Active
Radius Accounting ID: 2
Session ID: 2
Login Time: 2011-11-30 00:18:05 PST

Type: DHCP
IPv6 Prefix: 2041:1:1::/48
Logical System: default
Routing Instance: ASP-1
Interface: pp0.1073741825
Interface type: Static
MAC Address: 00:00:64:03:01:02
State: Active
Radius Accounting ID: jnpr :3
Session ID: 3
Underlying Session ID: 2
Login Time: 2011-11-30 00:18:35 PST
DHCP Options: len 42
00 08 00 02 0b b8 00 01 00 0a 00 03 00 01 00 00 64 03 01 02
00 06 00 02 00 19 00 19 00 0c 00 00 00 00 00 00 00 00 00 00
00 00
```

show subscribers detail (ACI Interface Set Session)

```
user@host> show subscribers detail
Type: VLAN
Logical System: default
Routing Instance: default
Interface: ge-1/0/0
Interface Set: aci-1001-ge-1/0/0.2800
Interface Set Session ID: 0
Underlying Interface: ge-1/0/0.2800
Dynamic Profile Name: aci-vlan-set-profile-2
Dynamic Profile Version: 1
State: Active
Session ID: 1
Agent Circuit ID: aci-ppp-dhcp-20
Login Time: 2012-05-26 01:54:08 PDT
```

show subscribers detail (PPPoE Subscriber Session with ACI Interface Set)

```
user@host> show subscribers detail
Type: PPPoE
User Name: ppphint2
IP Address: 10.10.1.5
Logical System: default
Routing Instance: default
Interface: pp0.1073741825
Interface type: Dynamic
Interface Set: aci-1001-demux0.1073741824
Interface Set Type: Dynamic
Interface Set Session ID: 2
Underlying Interface: demux0.1073741824
Dynamic Profile Name: aci-vlan-pppoe-profile
Dynamic Profile Version: 1
MAC Address: 00:00:64:39:01:02
State: Active
Radius Accounting ID: 3
```



```
Session ID: 3
Agent Circuit ID: aci-ppp-dhcp-dvlan-50
Login Time: 2012-03-07 13:46:53 PST
```

show subscribers extensive

```
user@host> show subscribers extensive
Type: DHCP
User Name: pd-user1
IPv6 Prefix: 2001:db8:db2:ffff:1::/64
Logical System: default
Routing Instance: default
Interface: ge-3/1/3.2
Interface type: Static
MAC Address: 00:51:ff:ff:00:03
State: Active
Radius Accounting ID: 1
Session ID: 1
Login Time: 2011-08-25 12:12:26 PDT
DHCP Options: len 42
00 08 00 02 00 00 00 01 00 0a 00 03 00 01 00 51 ff ff 00 03
00 06 00 02 00 19 00 19 00 0c 00 00 00 00 00 00 00 00 00 00
00 00
IPv6 Address Pool: pd_pool
IPv6 Network Prefix Length: 48
```

show subscribers extensive (RPF Check Fail Filter)

```
user@host> show subscribers extensive
...
Type: VLAN
  Logical System: default
  Routing Instance: default
  Interface: ae0.1073741824
  Interface type: Dynamic
  Dynamic Profile Name: vlan-prof
  State: Active
  Session ID: 9
  VLAN Id: 100
  Login Time: 2011-08-26 08:17:00 PDT
  IPv4 rpf-check Fail Filter Name: rpf-allow-dhcp
  IPv6 rpf-check Fail Filter Name: rpf-allow-dhcpv6
...
```

show subscribers extensive (L2TP LNS Subscribers on MX Series Routers)

```
user@host> show subscribers extensive
Type: L2TP
User Name: user1@example.net
IP Address: 10.1.32.58
IP Netmask: 255.255.0.0
Logical System: default
Routing Instance: default
Interface: si-5/2/0.1073749824
Interface type: Dynamic
Dynamic Profile Name: dyn-lns-profile2
Dynamic Profile Version: 1
State: Active
Radius Accounting ID: 8001
Session ID: 8001
Login Time: 2011-04-25 20:27:50 IST
```

```
IPv4 Input Filter Name: classify-si-5/2/0.1073749824-in
IPv4 Output Filter Name: classify-si-5/2/0.1073749824-out
```

show subscribers extensive (IPv4 and IPv6 Dual Stack)

```
user@host> show subscribers extensive
Type: VLAN
Logical System: default
Routing Instance: default
Interface: demux0.1073741824
Interface type: Dynamic
Dynamic Profile Name: svlanProfile
State: Active
Session ID: 1
Stacked VLAN Id: 0x8100.1001
VLAN Id: 0x8100.1
Login Time: 2011-11-30 00:18:04 PST

Type: PPPoE
User Name: dualstackuser1@EXAMPLE1.com
IP Address: 61.1.1.1
IPv6 Prefix: 2041:1:1::/48
IPv6 User Prefix: 2061:1:1:1::/64
Logical System: default
Routing Instance: ASP-1
Interface: pp0.1073741825
Interface type: Dynamic
Dynamic Profile Name: dualStack-Profile1
MAC Address: 00:00:64:03:01:02
State: Active
Radius Accounting ID: 2
Session ID: 2
Login Time: 2011-11-30 00:18:05 PST
IPv6 Delegated Network Prefix Length: 48
IPv6 Interface Address: 2061:1:1:1::1/64
IPv6 Framed Interface Id: 1:1:2:2
IPv4 Input Filter Name: FILTER-IN-pp0.1073741825-in
IPv4 Output Filter Name: FILTER-OUT-pp0.1073741825-out
IPv6 Input Filter Name: FILTER-IN6-pp0.1073741825-in
IPv6 Output Filter Name: FILTER-OUT6-pp0.1073741825-out

Type: DHCP
IPv6 Prefix: 2041:1:1::/48
Logical System: default
Routing Instance: ASP-1
Interface: pp0.1073741825
Interface type: Static
MAC Address: 00:00:64:03:01:02
State: Active
Radius Accounting ID: jnpr :3
Session ID: 3
Underlying Session ID: 2
Login Time: 2011-11-30 00:18:35 PST
DHCP Options: len 42
00 08 00 02 0b b8 00 01 00 0a 00 03 00 01 00 00 64 03 01 02
00 06 00 02 00 19 00 19 00 0c 00 00 00 00 00 00 00 00 00 00
00 00
IPv6 Delegated Network Prefix Length: 48
```


show subscribers extensive (Effective Shaping-Rate)

```
user@host> show subscribers extensive
Type: VLAN
Logical System: default
Routing Instance: default
Interface: demux0.1073741837
Interface type: Dynamic
Interface Set: ifset-1
Underlying Interface: ae1
Dynamic Profile Name: svlan-dhcp-test
State: Active
Session ID: 1
Stacked VLAN Id: 0x8100.201
VLAN Id: 0x8100.201
Login Time: 2011-11-30 00:18:04 PST
Effective shaping-rate: 31000000k
...
```

show subscribers aci-interface-set-name detail (Subscriber Sessions Using Specified ACI Interface Set)

```
user@host> show subscribers aci-interface-set-name aci-1003-ge-1/0/0.4001 detail
Type: VLAN
Logical System: default
Routing Instance: default
Interface: ge-1/0/0.
Underlying Interface: ge-1/0/0.4001
Dynamic Profile Name: aci-vlan-set-profile
Dynamic Profile Version: 1
State: Active
Session ID: 13
Agent Circuit ID: aci-ppp-vlan-10
Login Time: 2012-03-12 10:41:56 PDT

Type: PPPoE
User Name: ppphint2
IP Address: 10.10.1.7
Logical System: default
Routing Instance: default
Interface: pp0.1073741834
Interface type: Dynamic
Interface Set: aci-1003-ge-1/0/0.4001
Interface Set Type: Dynamic
Interface Set Session ID: 13
Underlying Interface: ge-1/0/0.4001
Dynamic Profile Name: aci-vlan-pppoe-profile
Dynamic Profile Version: 1
MAC Address: 00:00:65:26:01:02
State: Active
Radius Accounting ID: 14
Session ID: 14
Agent Circuit ID: aci-ppp-vlan-10
Login Time: 2012-03-12 10:41:57 PDT
```

show subscribers agent-circuit-identifier detail (Subscriber Sessions Using Specified ACI Substring)

```
user@host> show subscribers agent-circuit-identifier aci-ppp-vlan detail
Type: VLAN
Logical System: default
Routing Instance: default
Interface: ge-1/0/0.
```

```
Underlying Interface: ge-1/0/0.4001
Dynamic Profile Name: aci-vlan-set-profile
Dynamic Profile Version: 1
State: Active
Session ID: 13
Agent Circuit ID: aci-ppp-vlan-10
Login Time: 2012-03-12 10:41:56 PDT

Type: PPPoE
User Name: ppphint2
IP Address: 10.10.1.7
Logical System: default
Routing Instance: default
Interface: pp0.1073741834
Interface type: Dynamic
Interface Set: aci-1003-ge-1/0/0.4001
Interface Set Type: Dynamic
Interface Set Session ID: 13
Underlying Interface: ge-1/0/0.4001
Dynamic Profile Name: aci-vlan-pppoe-profile
Dynamic Profile Version: 1
MAC Address: 00:00:65:26:01:02
State: Active
Radius Accounting ID: 14
Session ID: 14
Agent Circuit ID: aci-ppp-vlan-10
Login Time: 2012-03-12 10:41:57 PDT
```

show subscribers interface extensive

```
user@host> show subscribers interface demux0.1073741826 extensive
Type: VLAN
User Name: test1@test.com
Logical System: default
Routing Instance: testnet
Interface: demux0.1073741826
Interface type: Dynamic
Dynamic Profile Name: profile-vdemux-relay-23qos
MAC Address: 00:00:6e:56:01:04
State: Active
Radius Accounting ID: 12
Session ID: 12
Stacked VLAN Id: 0x8100.1500
VLAN Id: 0x8100.2902
Login Time: 2011-10-20 16:21:59 EST

Type: DHCP
User Name: test1@test.com
IP Address: 172.16.200.6
IP Netmask: 255.255.255.0
Logical System: default
Routing Instance: testnet
Interface: demux0.1073741826
Interface type: Static
MAC Address: 00:00:6e:56:01:04
State: Active
Radius Accounting ID: 21
Session ID: 21
Login Time: 2011-10-20 16:24:33 EST
Service Sessions: 2
```



```
Service Session ID: 25
Service Session Name: SUB-QOS
State: Active

Service Session ID: 26
Service Session Name: service-cb-content
State: Active
IPv4 Input Filter Name: content-cb-in-demux0.1073741826-in
IPv4 Output Filter Name: content-cb-out-demux0.1073741826-out
```

show subscribers logical-system terse

```
user@host> show subscribers logical-system test1 terse
Interface           IP Address/VLAN ID   User Name         LS:RI
demux0.1073741825     11.0.0.3            RETAILER1-CLIENT  test1:retailer1
demux0.1073741826     12.0.0.3            RETAILER2-CLIENT  test1:retailer2
```

show subscribers physical-interface count

```
user@host> show subscribers physical-interface ge-1/0/0 count
Total subscribers: 3998, Active Subscribers: 3998
```

show subscribers routing-instance inst1 count

```
user@host> show subscribers routing-instance inst1 count
Total Subscribers: 188, Active Subscribers: 183
```

show subscribers stacked-vlan-id detail

```
user@host> show subscribers stacked-vlan-id 101 detail
Type: VLAN
Interface: ge-1/2/0.1073741824
Interface type: Dynamic
Dynamic Profile Name: svlan-prof
State: Active
Stacked VLAN Id: 0x8100.101
VLAN Id: 0x8100.100
Login Time: 2009-03-27 11:57:19 PDT
```

show subscribers stacked-vlan-id vlan-id detail (Combined Output)

```
user@host> show subscribers stacked-vlan-id 101 vlan-id 100 detail
Type: VLAN
Interface: ge-1/2/0.1073741824
Interface type: Dynamic
Dynamic Profile Name: svlan-prof
State: Active
Stacked VLAN Id: 0x8100.101
VLAN Id: 0x8100.100
Login Time: 2009-03-27 11:57:19 PDT
```

show subscribers stacked-vlan-id vlan-id interface detail (Combined Output for a Specific Interface)

```
user@host> show subscribers stacked-vlan-id 101 vlan-id 100 interface ge-1/2/0.* detail
Type: VLAN
Interface: ge-1/2/0.1073741824
Interface type: Dynamic
Dynamic Profile Name: svlan-prof
State: Active
Stacked VLAN Id: 0x8100.101
VLAN Id: 0x8100.100
Login Time: 2009-03-27 11:57:19 PDT
```


show subscribers user-name detail

```
user@host> show subscribers user-name larry1 detail
Type: DHCP
User Name: larry1
IP Address: 100.0.0.37
IP Netmask: 255.255.0.0
Logical System: default
Routing Instance: default
Interface: ge-1/0/0.1
Interface type: Static
Dynamic Profile Name: foo
MAC Address: 00:10:94:00:00:01
State: Active
Radius Accounting ID: 1
Session ID: 1
Login Time: 2011-11-07 08:25:59 PST
DHCP Options: len 52
35 01 01 39 02 02 40 3d 07 01 00 10 94 00 00 01 33 04 00 00
00 3c 0c 15 63 6c 69 65 6e 74 5f 50 6f 72 74 20 2f 2f 32 2f
37 2d 30 2d 30 37 05 01 06 0f 21 2c
```

show subscribers vlan-id

```
user@host> show subscribers vlan-id 100
Interface           IP Address                              User Name
ge-1/0/0.1073741824
ge-1/2/0.1073741825
```

show subscribers vlan-id detail

```
user@host> show subscribers vlan-id 100 detail
Type: VLAN
Interface: ge-1/0/0.1073741824
Interface type: Dynamic
Dynamic Profile Name: vlan-prof-tpid
State: Active
VLAN Id: 100
Login Time: 2009-03-11 06:48:54 PDT

Type: VLAN
Interface: ge-1/2/0.1073741825
Interface type: Dynamic
Dynamic Profile Name: vlan-prof-tpid
State: Active
VLAN Id: 100
Login Time: 2009-03-11 06:48:54 PDT
```

show subscribers vpi vci extensive (PPPoE-over-ATM Subscriber Session)

```
user@host> show subscribers vpi 40 vci 50 extensive
Type: PPPoE
User Name: testuser
IP Address: 100.0.0.2
IP Netmask: 255.255.0.0
Logical System: default
Routing Instance: default
Interface: pp0.0
Interface type: Static
MAC Address: 00:00:65:23:01:02
State: Active
```



```
Radius Accounting ID: 2
Session ID: 2
ATM VPI: 40
ATM VCI: 50
Login Time: 2012-12-03 07:49:26 PST
IP Address Pool: pool_1
IPv6 Framed Interface Id: 200:65ff:fe23:102
```

show subscribers summary

Syntax

show subscribers summary
<all>
< detail | extensive | terse>
<count>
<physical-interface *physical-interface-name*>
<logical-system *logical-system* pic | port | routing-instance *routing-instance* | slot>

Release Information Command introduced in Junos OS Release 10.2.

Description Display summary information for subscribers.

Options

all—(Optional) Display full subscriber summary.

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

count—(Optional) Display the count of total subscribers and active subscribers for any specified option.

logical-system—(Optional) Display subscribers whose logical system matches the specified logical system.

physical-interface-name—(M120, M320, and MX Series routers only) (Optional) Display a count of subscribers whose physical interface matches the specified physical interface, by subscriber state, client type and LS:RI.

pic—(M120, M320, and MX Series routers only) (Optional) Display a count of subscribers by PIC number and the total number of subscribers.

port—(M120, M320, and MX Series routers only) (Optional) Display a count of subscribers by port number and the total number of subscribers.

routing-instance—(Optional) Display subscribers whose routing instance matches the specified routing instance.

slot—(M120, M320, and MX Series routers only) (Optional) Display a count of subscribers by FPC slot number and the total number of subscribers.

NOTE: Due to display limitations, logical system and routing instance output values are truncated when necessary.

Required Privilege Level view

Related Documentation

- show subscribers on page 757

List of Sample Output show subscribers summary on page 778

show subscribers summary all on page 778
show subscribers summary physical-interface on page 778
show subscribers summary physical-interface pic on page 779
show subscribers summary physical-interface port on page 779
show subscribers summary physical-interface slot on page 779
show subscribers summary pic on page 779
show subscribers summary pic (Aggregated Ethernet Interfaces) on page 780
show subscribers summary port on page 780
show subscribers summary slot on page 780
show subscribers summary terse on page 780

Output Fields Table 30 on page 777 lists the output fields for the **show subscribers** command. Output fields are listed in the approximate order in which they appear.

Table 30: show subscribers summary Output Fields

Field Name	Field Description
Subscribers by State	Number of subscribers summarized by state. The summary information includes the following: • Init—Number of subscriber currently in the initialization state. • Configured—Number of configured subscribers. • Active—Number of active subscribers. • Terminating—Number of subscribers currently terminating. • Terminated—Number of terminated subscribers. • Total—Total number of subscribers for all states.
Subscribers by Client Type	Number of subscribers summarized by client type. Client types can include DHCP, L2TP, PPP, PPPOE, STATIC-INTERFACE, and VLAN. Also displays the total number of subscribers for all client types (Total).
Subscribers by LS:RI	Number of subscribers summarized by logical system:routing instance (LS:RI) combination. Also displays the total number of subscribers for all LS:RI combinations (Total).
Interface	Interface associated with the subscriber. The router or switch displays subscribers whose interface matches or begins with the specified interface. The * character indicates a continuation of addresses for the same session. For aggregated Ethernet interfaces, the output of the **summary (pic | port | slot)** options prefixes the interface name with **ae0:**.
Count	Count of subscribers displayed for each PIC, port, or slot when those options are specified with the **summary** option. For an aggregated Ethernet configuration, the total subscriber count does not equal the sum of the individual PIC, port, or slot counts, because each subscriber can be in more than one aggregated Ethernet link.
Total Subscribers	Total number of subscribers for all physical interfaces, all PICS, all ports, or all LS:RI slots.
IP Address/VLAN ID	Subscriber IP address or VLAN ID associated with the subscriber in the form *tpid.vlan-id*
User Name	Name of subscriber.
LS:RI	Logical system and routing instance associated with the subscriber.

Sample Output

show subscribers summary

```
user@host> show subscribers summary

Subscribers by State
 Init       3
 Configured     2
 Active   183
 Terminating     2
 Terminated     1

 TOTAL        191

Subscribers by Client Type
 DHCP     107
 PPP       76
 VLAN        8

 TOTAL        191
```

show subscribers summary all

```
user@host> show subscribers summary all
Subscribers by State
 Init          3
 Configured       2
 Active      183
 Terminating       2
 Terminated       1

 TOTAL          191

Subscribers by Client Type
 DHCP      107
 PPP          76
 VLAN          8

 TOTAL         191

Subscribers by LS:RI
 default:default   1
 default:ri1     28
 default:ri2     16
 ls1:default    22
 ls1:riA     38
 ls1:riB     44
 logsysX:routinstY  42

 TOTAL     191
```

show subscribers summary physical-interface

```
user@host> show subscribers summary physical-interface ge-1/0/0
Subscribers by State
   Active: 3998
   Total: 3998

Subscribers by Client Type
   DHCP: 3998
```

```
   Total: 3998

Subscribers by LS:RI
   default:default: 3998
   Total: 3998
```

show subscribers summary physical-interface pic

```
user@host> show subscribers summary physical-interface ge-0/2/0 pic
Subscribers by State
   Active: 4825
   Total: 4825

Subscribers by Client Type
   DHCP: 4825
   Total: 4825

Subscribers by LS:RI
   default:default: 4825
   Total: 4825
```

show subscribers summary physical-interface port

```
user@host> show subscribers summary physical-interface ge-0/3/0 port
Subscribers by State
   Active: 4825
   Total: 4825

Subscribers by Client Type
   DHCP: 4825
   Total: 4825

Subscribers by LS:RI
   default:default: 4825
   Total: 4825
```

show subscribers summary physical-interface slot

```
user@host> show subscribers summary physical-interface ge-2/0/0 slot
Subscribers by State
   Active: 4825
   Total: 4825

Subscribers by Client Type
   DHCP: 4825
   Total: 4825

Subscribers by LS:RI
   default:default: 4825
   Total: 4825
```

show subscribers summary pic

```
user@host> show subscribers summary pic
Interface           Count
ge-1/0              1000
ge-1/3              1000

Total Subscribers: 2000
```


show subscribers summary pic (Aggregated Ethernet Interfaces)

```
user@host> show subscribers summary pic
Interface           Count
ae0: ge-1/0         801
ae0: ge-1/3         801

Total Subscribers: 801
```

show subscribers summary port

```
user@host> show subscribers summary port
Interface           Count
ge-1                2000

Total Subscribers: 2000
```

show subscribers summary slot

```
user@host> show subscribers summary slot
Interface           Count
ge-1                2000

Total Subscribers: 2000
```

show subscribers summary terse

```
user@host> show subscribers summary terse
Interface           IP Address/VLAN ID   User Name         LS:RI
ge-1/3/0.1073741824   100                                    default:default
demux0.1073741824     100.0.0.10           WHOLESALER-CLIENT default:default
demux0.1073741825     101.0.0.3            RETAILER1-CLIENT  test1:retailer1
demux0.1073741826     102.0.0.3            RETAILER2-CLIENT  test1:retailer2
```

PART 9

Index

Index

Symbols

A

G

H

I

K

L

M

V

