JUNIPER NETWORKS

Junos® OS

Ethernet Automatic Protection Switching Feature Guide for Routing Devices

Release
14.1

Published: 2014-05-15

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

Junos® OS Ethernet Automatic Protection Switching Feature Guide for Routing Devices
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

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 xi defines notice icons used in this guide.

Table 1: Notice Icons

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

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

Table 2: Text and Syntax Conventions

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

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

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

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

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

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

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

Documentation Feedback

We encourage you to provide feedback, comments, and suggestions so that we can improve the documentation. You can send your comments to **techpubs-comments@juniper.net**, or fill out the documentation feedback form at https://www.juniper.net/cgi-bin/docbugreport/. If you are using e-mail, be sure to include the following information with your comments:

- Document or topic name
- URL or page number
- Software release version (if applicable)

Requesting Technical Support

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

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

Self-Help Online Tools and Resources

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

- Find CSC offerings: http://www.juniper.net/customers/support/
- Search for known bugs: http://www2.juniper.net/kb/
- Find product documentation: http://www.juniper.net/techpubs/
- Find solutions and answer questions using our Knowledge Base: http://kb.juniper.net/
- Download the latest versions of software and review release notes: http://www.juniper.net/customers/csc/software/

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

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

Opening a Case with JTAC

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

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

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

PART 1

Overview

CHAPTER 1

Ethernet Automatic Protection Switching

Ethernet Automatic Protection Switching Overview

Ethernet automatic protection switching (APS) is a linear protection scheme designed to protect VLAN based Ethernet networks.

With Ethernet APS, a protected domain is configured with two paths, a working path and a protection path. Both working and protection paths can be monitored using an Operations Administration Management (OAM) protocol like Connectivity Fault Management (CFM). Normally, traffic is carried on the working path (that is, the working path is the active path), and the protection path is disabled. If the working path fails, its protection status is marked as degraded (DG) and APS switches the traffic to the protection path, then the protection path becomes the active path.

APS uses two modes of operation, linear 1+1 protection switching architecture and linear 1:1 protection switching architecture. The linear 1+1 protection switching architecture operates with either unidirectional or bidirectional switching. The linear 1:1 protection switching architecture operates with bidirectional switching.

In the linear 1+1 protection switching architecture, the normal traffic is copied and fed to both working and protection paths with a permanent bridge at the source of the protected domain. The traffic on the working and protection transport entities is transmitted simultaneously to the sink of the protected domain, where a selection between the working and protection transport entities is made.

In the linear 1:1 protection switching architecture, the normal traffic is transported on either the working path or on the protection path using a selector bridge at the source of the protection domain. The selector at the sink of the protected domain selects the entity that carries the normal traffic.

Unidirectional and Bidirectional Switching

Unidirectional switching utilizes fully independent selectors at each end of the protected domain. Bidirectional switching attempts to configure the two end points with the same bridge and selector settings, even for a unidirectional failure. Unidirectional switching can protect two unidirectional failures in opposite directions on different entities.

Selective and Merging Selectors

In the linear 1:1 protection switching architecture, where traffic is sent only on the active path, there are two different ways in which the egress direction (the direction out of the protected segment) data forwarding can act: selective selectors and merging selectors. A selective selector forwards only traffic that is received from both the paths regardless of which one is currently active. In other words, with a merging selector the selection of the currently active path only affects the ingress direction. Merging selectors minimize the traffic loss during a protection switch, but they do not guarantee the delivery of the data packets in order.

Revertive and Nonrevertive Switching

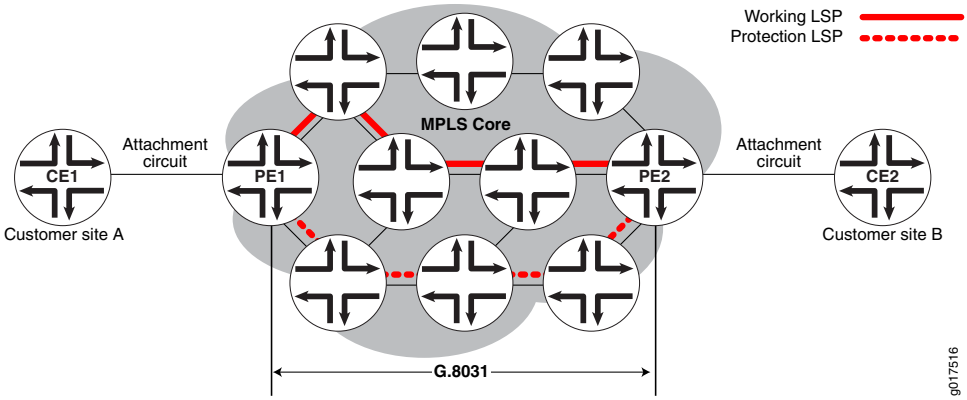
For revertive switching, traffic is restored to the working path after the conditions causing the switch have cleared.

For nonrevertive switching, traffic is allowed to remain on the protection path even after the conditions causing the switch have cleared.

NOTE: The configuration on both the provider edge (PE) routers have to be either in revertive mode or non-revertive mode.

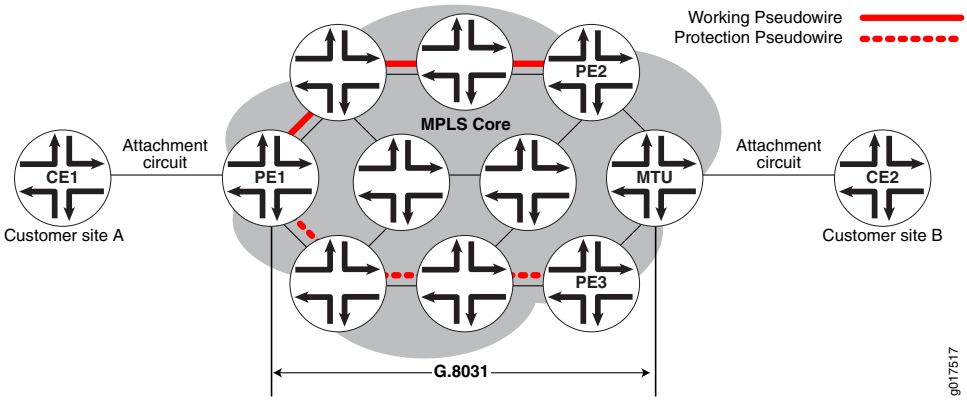
Protection Switching Between VPWS Pseudowires

Figure 1: Connections Terminating on Single PE



In the scenario diagramed in Figure 1 on page 4, a Virtual Private Wire Service (VPWS) is provisioned between customer sites A and B using a single pseudowire (layer 2 circuit) in the core network, and two Multiprotocol Label Switching (MPLS) Label Switched Paths (LSPs) are provisioned, one for the working path and the other one for the protection path. CFM CCM will be used to monitor the status of each LSP. Provider edge routers PE1 and PE2 run G.8031 Ethernet APS to select one of the LSPs as the active path. Once the active path is elected at the source end of the protection group, PE1 forwards to traffic from site A to the elected active path. At the sink end of the protection group, PE2 implements a merging selector, meaning it forwards the traffic coming from both the LSPs to the customer site B.

Figure 2: Connections Terminating on a Different PE



In the scenario represented in Figure 2 on page 5, a VPWS is provisioned between customer sites A and B using two pseudowires (layer 2 circuit) in the core network, one for the working path and the other for the protection path. CFM CCM will be used to monitor the status of each pseudowire.

Provider edge router PE1 and MTU run G.8031 Ethernet APS to select one of the pseudowires as the active path. Once the active path is elected at the source end of the protection group, PE1 forwards the traffic from site A to the elected active path. At the sink end of the protection group, MTU implements a merging selector, meaning it forwards the traffic coming from both the pseudowires to customer site B.

CLI Configuration Statements

```
[edit protocols protection-group]
ethernet-aps profile1{
   protocol g8031;
   revert-time seconds;
   hold-time 0-10000ms;
   local-request lockout;
}
```

revert-time- By default, protection logic restores the use of the working path once it recovers. The revert-time statement specifies how much time should elapse before the path for data should be switched from Protection to Working once recovery for Working has occurred. A revert-time of zero indicates no reversion. It will default to 300 sec (5 minutes) if not configured.

hold-time- Once a failure is detected, APS waits until this timer expires before initiating the protection switch. The range of the hold-time timer is 0 to 10,000 milliseconds. It will default to zero if not configured.

local-request- Configuring this value to lockout or force-switch will trigger lockout or force-switch operation on the protection groups using this profile.

Related Documentation

- Mapping of CCM Defects to APS Events on page 9
- Example: Configuring Protection Switching Between Psuedowires on page 10

PART 2

Configuration

CHAPTER 2

Ethernet Automatic Protection Switching

Mapping of CCM Defects to APS Events

The continuity check message (CCM) engine marks the status of working and protected transport entities as either Down, Degraded, or Up.

Down—The monitored path is declared down if any of the following Multiple End Point (MEP) defects occur:

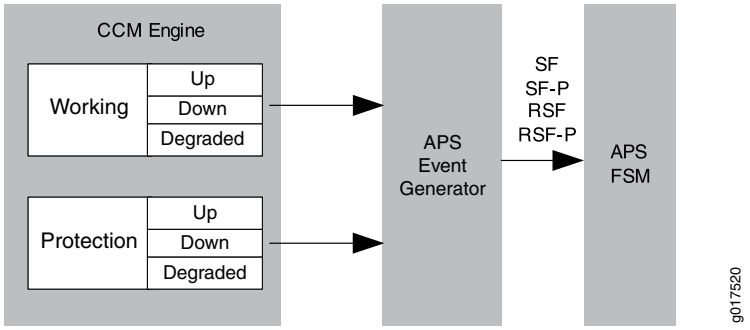
- Interface down
- CCM expiry
- RDI indicating signal failure

Degraded—The monitored path is declared degraded if any of the following MEP defects occur:

- FRR on
- FRR-ACK on

Up—The monitored path is declared up in the absence of any of the above events.

Figure 3: Understanding APS Events



As show in Figure 3 on page 9, the APS event generator generates the following APS events based on the status of the working and protection paths:

- **SF**—Signal failure on working path
- **RSF**—Working path recovers from signal failure
- **SF-P**—Signal failure on protection path
- **RSF-P**—Protection path recovers from signal failure

Related Documentation

- Ethernet Automatic Protection Switching Overview on page 3
- Example: Configuring Protection Switching Between Psuedowires on page 10

Example: Configuring Protection Switching Between Psuedowires

Requirements

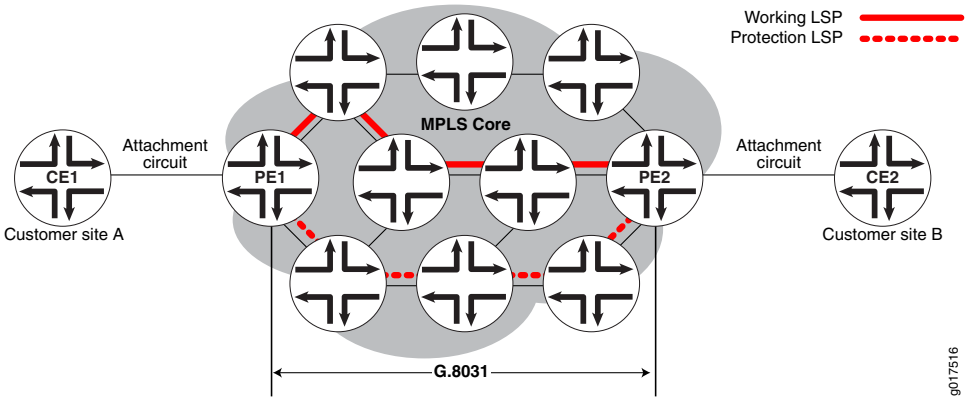
This example uses the following hardware and software components:

- Junos OS Release 11.2 or later
- 2 MX Series PE routers

Overview and Topology

The physical topology of the protection switching between psuedowires example is shown in Figure 4 on page 10.

Figure 4: Topology of a Network Using VPWS Psuedowires



The following definitions describe the meaning of the device abbreviations used in Figure 4 on page 10.

- Customer edge (CE) device—A device at the customer site that provides access to the service provider's VPN over a data link to one or more provider edge (PE) routers.

- Provider edge (PE) device—A device, or set of devices, at the edge of the provider network that presents the provider's view of the customer site.

Configuration

Step-by-Step Procedure

To configure protection switching between psuedowires, perform these tasks:

1. Configure automatic protection switching.

```
protocols {
  protection-group {
    ethernet-aps {
      profile-1 {
        protocol g8031;
        hold-time 1000s;
        revert-time 5m;
      }
    }
  }
```

2. Configure the connectivity fault management.

```
ethernet {
  oam {
    connectivity-fault-management {
      maintenance-domain md1 {
      level 5;
```

3. Configure the continuity check message for the working path.

```
maintenance-association W {
  protect maintenance-association P {
    aps-profile profile-1;
  }
  continuity-check {
    interval 1s;
  }
  mep 100 {
    interface ge-1/0/0.0 working;
    direction down;
    auto-discovery;
  }
}
```

4. Configure the continuity check message for the protection path.

```
maintenance-association P {
  continuity-check {
    interval 1s;
  }
  mep 100 {
    interface ge-1/0/0.0 protect;
    direction down;
    auto-discovery;
  }
}
```

Results Check the results of the configuration:

```
protocols {
  protection-group {
    ethernet-aps {
      profile-1 {
        protocol g8031;
        hold-time 1000s;
        revert-time 5m;
      }
    }
  }
  ethernet {
    oam {
      connectivity-fault-management {
        maintenance-domain md1 {
          level 5;
          maintenance-association W {
            protect maintenance-association P {
              aps-profile profile-1;
            }
            continuity-check {
              interval 1s;
            }
            mep 100 {
              interface ge-1/0/0.0 working;
              direction down;
              auto-discovery;
            }
          }
          maintenance-association P {
            continuity-check {
              interval 1s;
            }
            mep 100 {
              interface ge-1/0/0.0 protect;
              direction down;
              auto-discovery;
            }
          }
        }
      }
    }
  }
}
```

Related Documentation

- Ethernet Automatic Protection Switching Overview on page 3
- Mapping of CCM Defects to APS Events on page 9

CHAPTER 3

Network Interfaces Configuration Statements and Hierarchy

[edit protocols protection-group] Hierarchy Level

```
ethernet-ring ring-name {
  east-interface {
    control-channel channel-name {
      vlan number;
    }
  }
  guard-interval number;
  node-id mac-address;
  restore-interval number;
  ring-protection-link-owner;
  west-interface {
    control-channel channel-name {
      vlan number;
    }
  }
}
```

Related Documentation

- *Junos OS Hierarchy and RFC Reference*
- *Ethernet Interfaces*
- *Junos OS Network Interfaces Library for Routing Devices*

CHAPTER 4

Statement Summary

clear

Syntax request protection-group ethernet-aps clear md <md> ma <ma>

Hierarchy Level [edit protocols protection-group ethernet-aps]

Description Clears the lockout, force switch, manual switch, exercise, and wait-to-restore (WTR) states.

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

Related Documentation
- Ethernet Automatic Protection Switching Overview on page 3

exercise

Syntax request protection-group ethernet-aps exercise md <md> ma <ma>

Hierarchy Level [edit protocols protection-group ethernet-aps]

Description This configuration statement is used to test if APS is operating correctly, it does not interrupt regular APS operations.

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

Related Documentation
- Ethernet Automatic Protection Switching Overview on page 3

fast-aps-switch

Syntax fast-aps-switch;

Hierarchy Level [edit interfaces *interface-name* sonet-options aps]

Release Information Statement introduced in Junos OS Release 12.1.

Description (M320 routers with Channelized OC3/STM1 Circuit Emulation PIC with SFP only and EX Series switches) Reduce the Automatic Protection Switching (APS) switchover time in Layer 2 circuits.

NOTE:

- Configuring this statement reduces the APS switchover time only when the Layer 2 circuit encapsulation type for the interface receiving traffic from a Layer 2 circuit neighbor is SAToP.
- When the **fast-aps-switch** statement is configured in revertive APS mode, you must configure an appropriate value for revert time to achieve reduction in APS switchover time.
- To prevent the logical interfaces in the data path from being shut down, configure appropriate hold-time values on all the interfaces in the data path that support TDM.
- The **fast-aps-switch** statement cannot be configured when the APS **annex-b** option is configured.
- The interfaces that have the **fast-aps-switch** statement configured cannot be used in virtual private LAN service (VPLS) environments.

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

Related Documentation

- *Reducing APS Switchover Time in Layer 2 Circuits*

force switch

Syntax request protection-group ethernet-aps force-switch md <md> ma <ma>

Hierarchy Level [edit protocols protection-group ethernet-aps]

Description Forces traffic to switch from the active path to the alternate path. If the working path is the active path, traffic will be switched to the protection path. If the protection path is the active path, traffic will be switched to the protection path.

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

Related Documentation
- Ethernet Automatic Protection Switching Overview on page 3

lockout

Syntax request protection-group ethernet-aps lockout md <md> ma <ma>

Hierarchy Level [edit protocols protection-group ethernet-aps]

Description Configure a lockout of the protection path, forcing the use of the working path and locking out the protect path regardless of anything else.

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

Related Documentation
- Ethernet Automatic Protection Switching Overview on page 3

manual switch

Syntax request protection-group ethernet-aps manual-switch md <md> ma <ma>

Hierarchy Level [edit protocols protection-group ethernet-aps]

Description Forces traffic to switch from the active path to the alternate path, even in the absence of a failure on the working path. If the working path is the active path, traffic will be switched to the protection path. If the protection path is the active path, traffic will be switched to the protection path.

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

Related Documentation
- Ethernet Automatic Protection Switching Overview on page 3

PART 3

Troubleshooting

CHAPTER 5

Interface Diagnostics

Interface Diagnostics

You can use two diagnostic tools to test the physical layer connections of interfaces: loopback testing and bit error rate test (BERT) testing. Loopback testing enables you to verify the connectivity of a circuit. BERT testing enables you to identify poor signal quality on a circuit. This section contains the following topics:

Configuring Loopback Testing

Loopback testing allows you to verify the connectivity of a circuit. You can configure any of the following interfaces to execute a loopback test: Aggregated Ethernet, Fast Ethernet, Gigabit Ethernet, E1, E3, *N*xDS0, serial, SONET/SDH, T1, and T3.

The physical path of a network data circuit usually consists of segments interconnected by devices that repeat and regenerate the transmission signal. The transmit path on one device connects to the receive path on the next device. If a circuit fault occurs in the form of a line break or a signal corruption, you can isolate the problem by using a loopback test. Loopback tests allow you to isolate segments of the circuit and test them separately.

To do this, configure a *line loopback* on one of the routers. Instead of transmitting the signal toward the far-end device, the line loopback sends the signal back to the originating router. If the originating router receives back its own data link layer packets, you have verified that the problem is beyond the originating router. Next, configure a line loopback farther away from the local router. If this originating router does not receive its own data link layer packets, you can assume the problem is on one of the segments between the local router and the remote router's interface card. In this case, the next troubleshooting step is to configure a line loopback closer to the local router to find the source of the problem.

There are several types of loopback testing supported by the Junos OS, as follows:

- DCE local—Loops packets back on the local DCE.
- DCE remote—Loops packets back on the remote DCE.

- Local—Useful for troubleshooting physical PIC errors. Configuring local loopback on an interface allows transmission of packets to the channel service unit (CSU) and then to the circuit toward the far-end device. The interface receives its own transmission, which includes data and timing information, on the local router's PIC. The data received from the CSU is ignored. To test a local loopback, issue the **show interfaces *interface-name*** command. If PPP keepalives transmitted on the interface are received by the PIC, the **Device Flags** field contains the output **Loop-Detected**.
- Payload—Useful for troubleshooting the physical circuit problems between the local router and the remote router. A payload loopback loops data only (without clocking information) on the remote router's PIC. With payload loopback, overhead is recalculated.
- Remote—Useful for troubleshooting the physical circuit problems between the local router and the remote router. A remote loopback loops packets, including both data and timing information, back on the remote router's interface card. A router at one end of the circuit initiates a remote loopback toward its remote partner. When you configure a remote loopback, the packets received from the physical circuit and CSU are received by the interface. Those packets are then retransmitted by the PIC back toward the CSU and the circuit. This loopback tests all the intermediate transmission segments.

Table 3 on page 22 shows the loopback modes supported on the various interface types.

Table 3: Loopback Modes by Interface Type

Interface	Loopback Modes	Usage Guidelines
Aggregated Ethernet, Fast Ethernet, Gigabit Ethernet	Local	*Configuring Ethernet Loopback Capability*
Circuit Emulation E1	Local and remote	*Configuring E1 Loopback Capability*
Circuit Emulation T1	Local and remote	*Configuring T1 Loopback Capability*
E1 and E3	Local and remote	*Configuring E1 Loopback Capability* and *Configuring E3 Loopback Capability*
*N*xDS0	Payload	*Configuring Channelized E1 IQ and IQE Interfaces*, *Configuring T1 and NxDS0 Interfaces*, *Configuring Channelized OC12/STM4 IQ and IQE Interfaces (SONET Mode)*, *Configuring Channelized STM1 IQ and IQE Interfaces*, and *Configuring Channelized T3 IQ Interfaces*
Serial (V.35 and X.21)	Local and remote	*Configuring Serial Loopback Capability*
Serial (EIA-530)	DCE local, DCE remote, local, and remote	*Configuring Serial Loopback Capability*
SONET/SDH	Local and remote	*Configuring SONET/SDH Loopback Capability*

Table 3: Loopback Modes by Interface Type *(continued)*

Interface	Loopback Modes	Usage Guidelines
T1 and T3	Local, payload, and remote	*Configuring T1 Loopback Capability* and *Configuring T3 Loopback Capability* See also *Configuring the T1 Remote Loopback Response*

To configure loopback testing, include the **loopback** statement:

loopback *mode*;

You can include this statement at the following hierarchy levels:

- **[edit interfaces** *interface-name* **aggregated-ether-options]**
- **[edit interfaces** *interface-name* **ds0-options]**
- **[edit interfaces** *interface-name* **e1-options]**
- **[edit interfaces** *interface-name* **e3-options]**
- **[edit interfaces** *interface-name* **fastether-options]**
- **[edit interfaces** *interface-name* **gigether-options]**
- **[edit interfaces** *interface-name* **serial-options]**
- **[edit interfaces** *interface-name* **sonet-options]**
- **[edit interfaces** *interface-name* **t1-options]**
- **[edit interfaces** *interface-name* **t3-options]**

Interface Diagnostics

BERT allows you to troubleshoot problems by checking the quality of links. You can configure any of the following interfaces to execute a BERT when the interface receives a request to run this test: E1, E3, T1, T3; the channelized DS3, OC3, OC12, and STM1 interfaces; and the channelized DS3 IQ, E1 IQ, and OC12 IQ interfaces.

A BERT test requires a line loop to be in place on either the transmission devices or the far-end router. The local router generates a known bit pattern and sends it out the transmit path. The received pattern is then verified against the sent pattern. The higher the bit error rate of the received pattern, the worse the noise is on the physical circuit. As you move the position of the line loop increasingly downstream toward the far-end router, you can isolate the troubled portion of the link.

To configure BERT, you must configure the duration of the test, the bit pattern to send on the transmit path, and the error rate to monitor when the inbound pattern is received.

To configure the duration of the test, the pattern to send in the bit stream, and the error rate to include in the bit stream, include the **bert-period**, **bert-algorithm**, and **bert-error-rate** statements, respectively, at the **[edit interfaces** *interface-name interface-type*-**options]** hierarchy level:

[edit interfaces *interface-name interface-type*-options]
bert-algorithm *algorithm*;
bert-error-rate *rate*;
bert-period *seconds*;

By default, the BERT period is 10 seconds. You can configure the BERT period to last from 1 through 239 seconds on some PICs and from 1 through 240 seconds on other PICs.

rate is the bit error rate. This can be an integer from 0 through 7, which corresponds to a bit error rate from 10^{-0} (1 error per bit) to 10^{-7} (1 error per 10 million bits).

algorithm is the pattern to send in the bit stream. For a list of supported algorithms, enter a **?** after the **bert-algorithm** statement; for example:

```
[edit interfaces t1-0/0/0 t1-options]

user@host# set bert-algorithm ?
Possible completions:
  pseudo-2e11-o152     Pattern is 2^11 -1 (per O.152 standard)
  pseudo-2e15-o151     Pattern is 2^15 - 1 (per O.152 standard)
  pseudo-2e20-o151     Pattern is 2^20 - 1 (per O.151 standard)
  pseudo-2e20-o153     Pattern is 2^20 - 1 (per O.153 standard)
        ...
```

For specific hierarchy information, see the individual interface types.

NOTE: The 4-port E1 PIC supports only the following algorithms:

```
pseudo-2e11-o152     Pattern is 2^11 -1 (per O.152 standard)
pseudo-2e15-o151     Pattern is 2^15 - 1 (per O.151 standard)
pseudo-2e20-o151     Pattern is 2^20 - 1 (per O.151 standard)
pseudo-2e23-o151     Pattern is 2^23 (per O.151 standard)
```

When you issue the help command from the CLI, all BERT algorithm options are displayed, regardless of the PIC type, and no commit check is available. Unsupported patterns for a PIC type can be viewed in system log messages.

NOTE: The 12-port T1/E1 Circuit Emulation (CE) PIC supports only the following algorithms:

```
all-ones-repeating   Repeating one bits
all-zeros-repeating  Repeating zero bits
alternating-double-ones-zeros  Alternating pairs of ones and zeros
alternating-ones-zeros  Alternating ones and zeros
pseudo-2e11-o152     Pattern is 2^11 -1 (per O.152 standard)
pseudo-2e15-o151     Pattern is 2^15 - 1 (per O.151 standard)
pseudo-2e20-o151     Pattern is 2^20 - 1 (per O.151 standard)
pseudo-2e7           Pattern is 2^7 - 1
pseudo-2e9-o153      Pattern is 2^9 - 1 (per O.153 standard)
repeating-1-in-4     1 bit in 4 is set
repeating-1-in-8     1 bit in 8 is set
repeating-3-in-24    3 bits in 24 are set
```

When you issue the help command from the CLI, all BERT algorithm options are displayed, regardless of the PIC type, and no commit check is available. Unsupported patterns for a PIC type can be viewed in system log messages.

NOTE: The IQE PICs support only the following algorithms:

```
all-ones-repeating   Repeating one bits
all-zeros-repeating  Repeating zero bits
alternating-double-ones-zeros  Alternating pairs of ones and zeros
alternating-ones-zeros  Alternating ones and zeros
pseudo-2e9-o153      Pattern is 2^9  -1 (per O.153 (511 type) standard)
pseudo-2e11-o152     Pattern is 2^11 -1 (per O.152 and O.153 (2047 type)
 standards)
pseudo-2e15-o151     Pattern is 2^15 -1 (per O.151 standard)
pseudo-2e20-o151     Pattern is 2^20 -1 (per O.151 standard)
pseudo-2e20-o153     Pattern is 2^20 -1 (per O.153 standard)
pseudo-2e23-o151     Pattern is 2^23 -1 (per O.151 standard)
repeating-1-in-4     1 bit in 4 is set
repeating-1-in-8     1 bit in 8 is set
repeating-3-in-24    3 bits in 24 are set
```

When you issue the help command from the CLI, all BERT algorithm options are displayed, regardless of the PIC type, and no commit check is available. Unsupported patterns for a PIC type can be viewed in system log messages.

NOTE: BERT is supported on the PDH interfaces of the Channelized SONET/SDH OC3/STM1 (Multi-Rate) MIC with SFP and the DS3/E3 MIC. The following BERT algorithms are supported:

```
all-ones-repeating             Repeating one bits
all-zeros-repeating            Repeating zero bits
alternating-double-ones-zeros  Alternating pairs of ones and zeros
alternating-ones-zeros         Alternating ones and zeros
repeating-1-in-4               1 bit in 4 is set
repeating-1-in-8               1 bit in 8 is set
repeating-3-in-24              3 bits in 24 are set
pseudo-2e9-o153                Pattern is 2^9 - 1 (per 0.153 standard)
pseudo-2e11-o152               Pattern is 2^11 - 1 (per 0.152 standard)
pseudo-2e15-o151               Pattern is 2^15 - 1 (per 0.151 standard)
pseudo-2e20-o151               Pattern is 2^20 - 1 (per 0.151 standard)
pseudo-2e20-o153               Pattern is 2^20 - 1 (per 0.153 standard)
pseudo-2e23-o151               Pattern is 2^23 (per 0.151 standard)
```

Table 4 on page 26 shows the BERT capabilities for various interface types.

Table 4: BERT Capabilities by Interface Type

Interface	T1 BERT	T3 BERT	Comments
12-port T1/E1 Circuit Emulation	Yes (ports 0–11)		• Limited algorithms
4-port Channelized OC3/STM1 Circuit Emulation	Yes (port 0–3)		• Limited algorithms
E1 or T1	Yes (port 0–3)	Yes (port 0–3)	• Single port at a time • Limited algorithms
E3 or T3	Yes (port 0–3)	Yes (port 0–3)	• Single port at a time
Channelized OC12	N/A	Yes (channel 0–11)	• Single channel at a time • Limited algorithms • No bit count
Channelized STM1	Yes (channel 0–62)	N/A	• Multiple channels • Only one algorithm • No error insert • No bit count
Channelized T3 and Multichannel T3	Yes (channel 0–27)	Yes (port 0–3 on channel 0)	• Multiple ports and channels • Limited algorithms for T1 • No error insert for T1 • No bit count for T1

These limitations do not apply to channelized IQ interfaces. For information about BERT capabilities on channelized IQ interfaces, see *Channelized IQ and IQE Interfaces Properties*.

Starting and Stopping a BERT Test

Before you can start the BERT test, you must disable the interface. To do this, include the **disable** statement at the **[edit interfaces *interface-name*]** hierarchy level:

```
[edit interfaces interface-name]
disable;
```

After you configure the BERT properties and commit the configuration, begin the test by issuing the **test interface *interface-name interface-type*-bert-start** operational mode command:

```
user@host> test interface interface-name interface-type-bert-start
```

The test runs for the duration you specify with the **bert-period** statement. If you wish to terminate the test sooner, issue the **test interface *interface-name interface-type*-bert-stop** command:

```
user@host> test interface interface-name interface-type-bert-stop
```

For example:

```
user@host> test interface t3-1/2/0 t3-bert-start
user@host> test interface t3-1/2/0 t3-bert-stop
```

To view the results of the BERT test, issue the **show interfaces extensive | find BERT** command:

```
user@host> show interfaces interface-name extensive | find BERT
```

For more information about running and evaluating the results of the BERT procedure, see the CLI Explorer.

NOTE: To exchange BERT patterns between a local router and a remote router, include the **loopback remote** statement in the interface configuration at the remote end of the link. From the local router, issue the **test interface** command.

Example: Configuring Bit Error Rate Testing

Configure a BERT test on a T3 interface. In this example, the run duration lasts for 120 seconds. The configured error rate is 0, which corresponds to a bit error rate of 10^{-0} (1 error per bit). The configured bit pattern of **all-ones-repeating** means that every bit the interface sends is a set to a value of 1.

```
[edit interfaces]
t3-1/2/0 {
    t3-options {
        bert algorithm all-ones-repeating;
        bert-error-rate 0;
        bert-period 120;
```

```
    }
}
```

PART 4

Index

Index

