Chapter 13

Managing Packet Mirroring

Packet mirroring enables you to send a copy of a packet to an external host for analysis. Packet mirroring has many uses, including traffic debugging and troubleshooting user networking problems.

This chapter contains the following topics:

- Avoiding Conflicts Between CLI-Based and RADIUS-Based Packet Mirroring Configurations on page 215
- Understanding the Prepended Header During a Packet Mirroring Session on page 216
- Resolving and Tracking the Analyzer Device's Address on page 219
- Using Multiple Triggers for CLI-Based Packet Mirroring on page 219
- Optimizing Packet Mirroring Performance on page 220
- Logging Packet Mirroring Information on page 222
- Using SNMP Secure Packet Mirroring Traps on page 222
- Capturing SNMP Secure Audit Logs on page 226

Avoiding Conflicts Between CLI-Based and RADIUS-Based Packet Mirroring Configurations

The JUNOSe software gives you a great deal of flexibility in creating your packet mirroring environment by supporting both the CLI-based and the RADIUS-based configuration methods. However, a conflict might occur when you use both methods. For example, you might have both a CLI-based session and a RADIUS-based session for the same subscriber, each session using a unique secure policy list.

To avoid potential conflicts when both CLI-based and RADIUS-based configurations exist for a subscriber, the JUNOSe software uses the following rules to determine which configuration to use:

- When a user logs in—The RADIUS-based configuration is always used
- When the user is already logged in—The new configuration always replaces the existing configuration, regardless of creation method.

Understanding the Prepended Header During a Packet Mirroring Session

During a packet mirroring session, the router prepends a special UDP/IP header to each mirrored packet that is sent to the analyzer interface. This prepended header is created by the policy-mirroring action, and is used for demultiplexing at the analyzer to sort through the multiple mirrored streams that arrive from different sources.

All mirrored L2TP session packets are prepended with UDP/IP header. However, for IP traffic mirroring, the prepend header is optional; the header is added if the mirroring-related VSAs (VSAs 59 and 61) are included in the RADIUS message. For CLI-based mirroring, the **analyzer-udp-port** keyword of the **mirror analyzer-ip-address** command creates the same information contained in the two VSAs. If you do not include the VSAs or the **analyzer-udp-port** keyword, an IP mirroring action is indicated, and the prepend header is not used.

NOTE: For IP mirroring, both VSA 26-59 and 26-61 or neither must be included. If only one of the VSAs is used, the configuration fails.

Figure 21 shows the structure of the prepended header. The values in parentheses indicate the fixed value for individual fields. For fields that do not have a fixed value listed, the value is dynamically created for each mirrored packet. Table 47 on page 217 lists the fields in the prepended header and indicates the values and field length.

Figure 21: Prepended Header

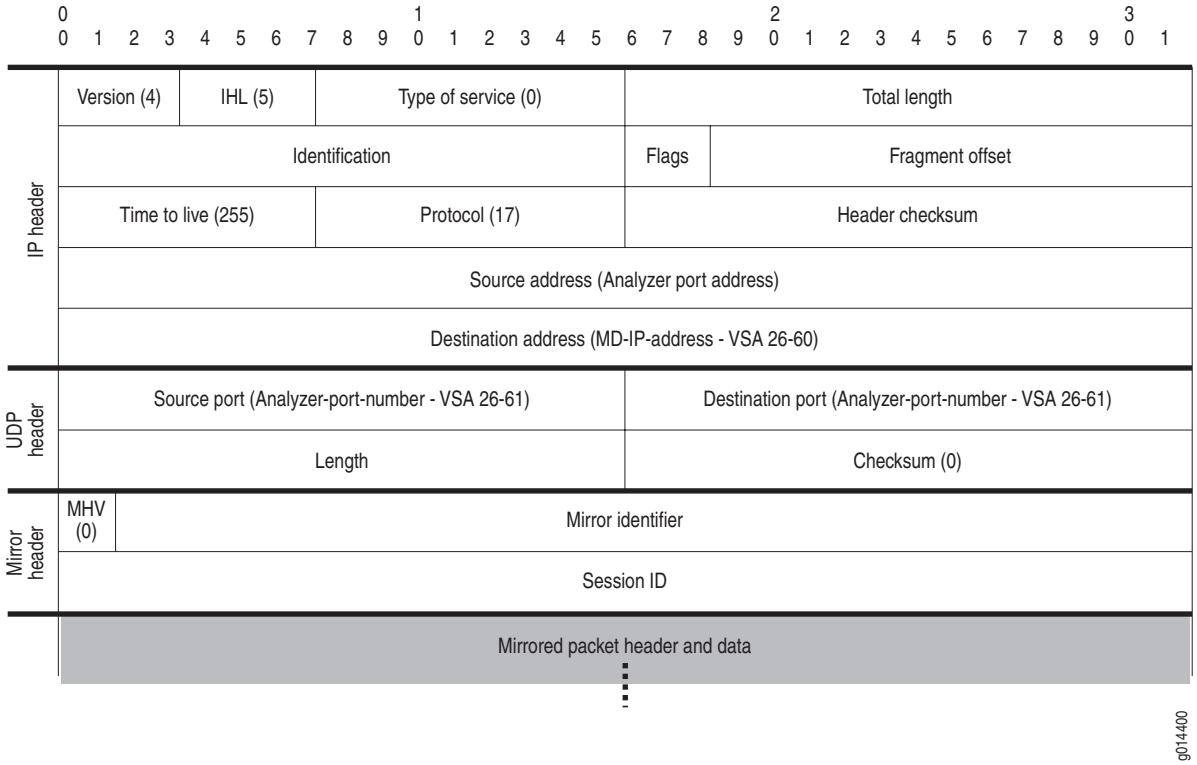

Table 47: Prepended Header Field Descriptions

Field	Value	Length (Bits)
IP Header		
Version	4	4
IHL	5	4
Type of Service	0	8
Total Length	Dynamically computed	16
Identification	Dynamically computed	16
Flags	Dynamically computed	3
Fragment Offset	Dynamically computed	13
Time to Live	255	8
Protocol	17	8
Header Checksum	Dynamically computed	16
Source Address	Analyzer interface IP address	32
Destination Address	VSA 26-60	32
UDP Header		
Source Port	VSA 26-61	16
Destination Port	VSA 26-61	16
Length	Dynamically computed	16
Checksum	0	16

Table 47: Prepended Header Field Descriptions (continued)

Field	Value	Length (Bits)
Mirror Header		
MHV (mirror header value)	0	2
Mirror Identifier	See *Format of the Mirror Header Attributes* on page 218 for details	30
Session-ID	See *Format of the Mirror Header Attributes* on page 218 for details	32

Format of the Mirror Header Attributes

The mirror header values are determined by the value that you configure in VSA 26-59. VSA 26-59 is declared as a hexadecimal string that can be either 8 bytes or 4 bytes long. The 8-byte format enables you to further specify the value that is used for the Session-ID field. If you use the 4-byte format, the router automatically determines the Session-ID field. The value in the 2-bit version field specifies the format that is used—0 indicates the 8-byte format, and 1 indicates the 4-byte format.

8-Byte Format

The 8-byte format of VSA 26-59 enables you to manually specify the Session-ID value in addition to the Mirror Identifier value. To use the 8-byte format, you configure the first two most significant bits of the first word of the VSA to a value of 0, which indicates two words in the VSA. The remaining 30 bits of the first word form the Mirror Identifier value, and the second word is the Session-ID field. You cannot change the order of these two words.

For example, a value of 0000030000000090 in VSA 26-59 configures the following fields in the mirror header, as shown in Figure 22:

- MHV = 0
- Mirror Identifier = 0x300
- Session-ID = 0x90

Figure 22: 8-Byte Format of VSA 26-59

4-Byte Format

To use the 4-byte format of VSA 26-59, you configure the first two most significant bits of the VSA to a value of 1, which indicates a single word in the VSA. The remaining 30 bits of the word form the Mirror Identifier value. The router then creates the Session-ID value based on the least significant 32 bits of the Acct-Session-ID (RADIUS attribute 44).

For example, a value of 40000010 for VSA 26-59 configures the following fields in the mirror header, as shown in Figure 23:

- MHV = 1
- Mirror Identifier = 0x10

Figure 23: 4-Byte Format of VSA 26-59

Resolving and Tracking the Analyzer Device's Address

During the packet mirroring configuration process, you specify the IP address of the analyzer device to which the mirrored traffic is sent. For CLI-based packet mirroring, you use the **mirror analyzer-ip-address** command to specify the IP address. For RADIUS-based packet mirroring, the RADIUS attribute Med-IP-Address [26-60] is the address of the analyzer device.

After configuration is complete, the router performs a route lookup to resolve the analyzer device's address and to ensure that traffic can be forwarded to the analyzer device for analysis. However, the analyzer device is considered unreachable if the router's analyzer interface is not in analyzer mode, is not yet created, or if the routes to the analyzer device are absent

If the analyzer device is unreachable, then the mirror action in the secure policy is disabled, and no packets are mirrored. The **show secure policy-list** command output indicates that the mirror action is disabled and the analyzer device is unreachable.

The router tracks the analyzer device's IP address for any route changes within the router. This tracking ability provides a degree of failure recovery by enabling you to configure multiple analyzer interfaces to serve as redundant ports to reach the analyzer device.

Using Multiple Triggers for CLI-Based Packet Mirroring

When you configure CLI-based packet mirroring, you can create multiple mirroring rules for a particular subscriber. For example. you might create two rules; one that uses IP address as the trigger that identifies the user and a second with the subscriber's username as the trigger. You can also configure RADIUS-based mirroring to use multiple methods to identify subscribers.

To avoid conflicts between multiple mirroring rules, both CLI-based and RADIUS based mirroring operations assign a precedence to the subscriber identification triggers. When multiple rules are configured for the same subscriber, the rule with the highest precedence is used to identify the subscriber.

The following list indicates the order of precedence for the subscriber identification triggers, with the acct-session-id having the highest precedence.

1. acct-session-id
2. calling-station-id
3. ip-address (virtual router specific)
4. nas-port-id
5. username (virtual router specific)

For example, if you create the following three rules for a subscriber, the packet mirroring session uses the rule with the acct-session-id to identify the subscriber. When there are multiple rules, if the selected rule fails, the router denies the packet mirroring request and does not attempt to use the other rules.

host1(config)#**mirror acct-session-id atm 2/1.2:0.42:0001048579 ip secure-policy-list securePolicyIp10**
host1(config)#**mirror ip-address 192.168.105.25 ip secure-policy-list securePolicyIp4**
host1(config)#**mirror username jwbooth@isptheatre.com ip secure-policy-list securePolicyIp15**

If the packet mirroring request is a RADIUS-initiated session (a RADIUS-based packet mirroring session for a subscriber who is already logged in), the router verifies the validity of all of the mirroring rules related to the particular subscriber. If any of the rules fail (for example, the identification fields do not match), the packet mirroring request is denied.

The calling-station-id trigger is externally visible only for tunneled users (if there are no RADIUS overrides). If a case-sensitive user name does not match a subscriber's name or if the dynamic IP interface UID does not exist, the subscriber is disregarded.

Optimizing Packet Mirroring Performance

Packet mirroring operations require some system resources. As a general rule, to avoid performance degradation, limit the amount of mirrored traffic to a maximum of 5 percent of the E-series router's total traffic.

For many packet mirroring environments, using the 5-percent guideline is sufficient. However, if you want to more closely manage packet mirroring's use of your router's resources, this section provides guidelines and equations to help you determine your packet mirroring requirements.

The guidelines for packet mirroring requirements use the following assumptions for a specific line module:

A = Total input traffic at the line module

B = Total output traffic at the line module

X = Amount of traffic mirrored at input in the line module

Y = Amount of traffic mirrored at output in the line module

Determine Traffic Loads

Using the previous assumptions, you can determine traffic loads for a given line module:

A =	Load at ingress side of the line module
(B + X) =	Load at egress side of the line module
(A + 2X + Y) =	Load at ingress to fabric from the line module

Establish Resource Guidelines

Next, using the traffic loads that you determined for the line module, you can establish guidelines for the amount of packet mirroring traffic for your router.

If you exceed these guidelines, regular (non-packet mirroring) packets from all subscribers, including nonmirrored subscribers, will be dropped. If the fabric bandwidth is not exceeded, then the performance penalties are contained within the slot where the packet mirroring activity occurs. However, if the fabric bandwidth is exceeded, traffic from other line modules might also be dropped.

- (A + 2X + Y) must be less than the maximum fabric bandwidth supported from this line module.
- (2X + Y) must be less than 100Mbps (the enforced queue limit).

  The 100 Mbps limit does not apply to the following line modules:

  - GE-2 line module (ERX-310 router and ERX-1440 router)
  - GE-HDE line module (ERX-310 router and ERX-1440 router)
  - OC48 Frame APS I/O module (ERX-1440 router only)
  - ES2 4G LM (E120 router and E320 router)
- (B + X) must be less than the maximum supported egress bandwidth.
- The number of mirrored interfaces per line module must be less than 1023 (the configuration enforced for secure policy attachments).
- The number of interfaces mirrored per chassis must be less than 2400 (the configuration enforced for secure policy attachments).

NOTE: Packet mirroring can also affect the forwarding controller's packet handling performance.

Logging Packet Mirroring Information

The JUNOSe software's packet mirroring feature provides two secure methods of capturing and displaying packet mirroring-related information. Both methods ensure security by requiring the **mirror-enable** command to be enabled.

- Secure logging—Captures packet mirroring information to a local secure log on the router.
- SNMP secure packet mirroring traps—Captures and reports packet mirroring information to an external device; you can then use the privileged **show mirror trap** and **show snmp traps** CLI commands to view secure trap configuration information.

  SNMP agent also implements a secure audit logging facility for the debugging of packet mirroring traps and packet Mirror-MIB accesses. When secure audit logging is enabled, SNMP agent logs reported mirror traps and packet Mirror-MIB get/set operations to local volatile memory on the router.

By default, the JUNOSe software captures packet mirroring-related activity to a secure local mirror log. No action is required on your part to enable or disable the logging process; however, only authorized users can access the secure log.

The secure logging feature includes the **clear mirror log** and **show mirror log** commands. The **mirror-enable** command must be enabled to make the commands visible in the CLI.

Related Topics

- **clear mirror log** command
- **show mirror log** command

Using SNMP Secure Packet Mirroring Traps

SNMP secure packet mirroring traps enable you to capture and report packet mirroring information to an external device; you can then view the secure information on the remote device. The secure packet mirroring traps feature is an extension of the router's standard SNMP implementation, and is only available to SNMPv3 users who are authorized to use packet mirroring.

You can also log mirror traps to local volatile memory for debugging purposes by enabling the SNMP secure log feature. See *Capturing SNMP Secure Audit Logs* on page 226 for details of secure audit logging. Normal console and syslog audit logs for packet mirroring traps and packet Mirror-MIB accesses are suppressed due to security concerns.

NOTE: The contents of secure logs are not preserved across a reboot.

The **mirror-enable** command must be enabled to make packet mirroring-related commands, command options, and **show** command output visible.

NOTE: You must use the CLI to configure the secure packet mirroring trap category to allow transmission of secure packet mirroring traps through the router—you cannot use SNMP to configure the secure packet mirroring trap category. However, after you have configured the secure packet mirroring trap category using the CLI, you can then use SNMP (juniPacketMirrorMIB.mi2) to enable and disable secure packet mirroring traps.

Related Topics

- See *JUNOSe System Basics Configuration Guide, Chapter 4, Configuring SNMP* for information about JUNOSe software SNMP support.
- **mirror trap-enable** command
- **snmp-server clear secure-log** command
- **snmp-server enable traps** command
- **snmp-server host** command
- **snmp-server secure-log** command
- **show mirror trap** command
- **show snmp secure-log** command

Table 48 indicates the events that trigger secure packet-mirroring traps and lists the information sent in the trap for each event.

Table 48: Packet-Mirroring SNMP Traps

Trap Information Sent	Event That Triggers the Trap			
	A secure policy failed during CoA-based or RADIUS-initiated packet mirroring	**A secure policy failed during CLI trigger or CLI-based packet mirroring**	**An interface with secure policies attached is deleted**	**An analyzer is unreachable**
Analyzer address	–	–	–	a
Application name	a	a	–	–
Configuration source	a	a	a	–
Date and time of event	–	a	a	a
Error cause	a	a	–	–
Error string	a	a	–	–
Mirror ID	a	–	a	–
Mirroring direction	–	–	a	–
Secure policy name	–	a	a	–

Table 48: Packet-Mirroring SNMP Traps (continued)

Trap Information Sent	Event That Triggers the Trap			
	A secure policy failed during CoA-based or RADIUS-initiated packet mirroring	A secure policy failed during CLI trigger or CLI-based packet mirroring	An interface with secure policies attached is deleted	An analyzer is unreachable
Secure policy UID	–	a	a	–
Session ID	a	–	a	–
Trigger event	a	a	a	–
Trigger type	a	a	a	–
Username	a	–	–	–
Virtual router (0 for L2TP)	a	a	a	a

Additional Packet-Mirroring Traps for CALEA Compliance

You can use the packet-mirroring traps shown in Table 49 to help support compliance with the Communications Assistance for Law Enforcement Act (CALEA), which defines electronic surveillance guidelines for telecommunications companies. For example, a third-party vendor of mediation devices might receive packet mirroring traps from the router and convert the traps to messages that comply with CALEA, such as Lawfully Authorized Electronic Surveillance (LAES) for IP Network Access, American Nation Standard For Telecommunications messages. Individual traps might map to multiple LAES messages to provide additional compliance-related information.

Table 49: Packet-Mirroring Traps for CALEA Compliance

Trap	Description
juniPacketMirrorSessionStart	A grant has been issued to a mirrored subscriber.
juniPacketMirrorSessionEnd	A mirrored session has been terminated; includes the termination reason.
juniPacketMirrorInterfaceSessionActivated	A secure policy has been attached to an existing interface or to an existing session.
juniPacketMirrorInterfaceSessionDeactivated	A secure policy has been detached from an interface, not including interface or session termination.
juniPacketMirrorSessionReject	A deny has been issued because the potential mirrored user was not allowed on the network for some reason. However, the user would have been mirrored if access to the network had been allowed.
juniPacketMirrorSessionFailed	The user session was terminated before the secure policy was attached. For example, no resources were available to create the interface. The termination reason is included.

Packet Mirroring Trap Severity Levels

Table 50 lists the default severity levels for packet mirroring traps. See Table 23 in *JUNOSe System Basics Configuration Guide, Chapter 4, Configuring SNMP* for descriptions of the severity levels.

Table 50: Packet Mirroring Trap Severity Levels

Trap	Default Severity Level
juniPacketMirrorAnalyzerUnreachable	Warning
juniPacketMirrorCliTriggerBasedMirroringFailure	Error
juniPacketMirrorInterfaceDeleted	Notice
juniPacketMirrorInterfaceSessionActivated	Info
juniPacketMirrorInterfaceSessionDeactivated	Info
juniPacketMirrorRadiusBasedMirroringFailure	Error
juniPacketMirrorSessionEnd	Info
juniPacketMirrorSessionFailed	Info
juniPacketMirrorSessionStart	Info
juniPacketMirrorSessionReject	Info

Configuring SNMP Secure Packet Mirroring Traps

To configure SNMP secure traps support, perform the following tasks on your E-series router:

1. Enable packet mirroring support.
2. Configure the packet mirroring application to generate traps.
3. (Optional) Verify the packet mirroring trap configuration.
4. (Optional) Configure the SNMP server to support secure logs.
5. Configure the SNMP server to generate packet mirroring traps.
6. Configure the SNMPv3 user for whom packet mirroring traps are generated.
7. Configure the SNMP server to report packet mirroring traps to a remote host.
8. (Optional) Verify the SNMP server packet mirroring configuration.

The following example illustrates the procedure to configure SNMP secure packet mirroring traps support:

```
host1#mirror-enable
host1#configure terminal
host1(config)#mirror trap-enable
host1(config)#show mirror trap
Traps are enabled
host1(config)#snmp-server secure-log
host1(config)#snmp-server user fredMirrorUser group mirror authentication md5
fred-md5password privacy des fred-despassword
```

```
host1(config)#snmp-server enable traps packetMirror trapFilters notice
host1(config)#snmp-server host 192.168.57.103 version 3 fredMirrorUser
cliSecurityAlert packetMirror trapFilters notice
host1(config)#show snmp trap

Enabled Categories: CliSecurity, PacketMirror, Sonet
SNMP authentication failure trap is disabled
Trap Source: FastEthernet 6/0, Trap Source Address:192.168.120.78
Trap Proxy: enabled
Global Trap Severity Level: 6 - informational

Address          Security String                   Ver  Port   Trap Categories
---------------  --------------------------------  ---  -----  ----------------
192.168.1.1      host1                             v1   162    Cli
192.168.57.103   fredMirrorUser                    v3   162    CliPacketMirror
192.168.57.162   host2                             v3   162    Sonet


Address         TrapSeverityFilter  Ping    Maximum    Queue    Queue Full
                                    TimeOut QueueSize DrainRate discrd methd
--------------- ------------------ ------- --------- --------- -------------
192.168.1.1     5 - notice         1       32        0         dropLastIn
192.168.57.103  5 - notice         1       32        0         dropLastIn
192.168.57.162  2 - critical       1       32        0         dropLastIn
```

Capturing SNMP Secure Audit Logs

SNMP secure audit logging enables administrators to collect the SNMP audit logs for mirror traps and Mirror-MIB get/set operations with the protection of the mirror enabling feature. Secure audit logging facilitates the debugging of issues related to SNMP packet mirror traps.

All normal SNMP console and syslog audit logs (including snmpTrap, snmpPduAudit, and snmpSetPduAudit) for secure traps and Mirror-MIB are suppressed due to security concerns. When you have issued the **mirror enable** command, you can issue the **snmp secure-log** command to capture secure audit logs. Configuration, storage, and display of the SNMP secure logging is on global basis rather than a per-VR basis.

The SNMP agent captures and stores the audit logs for secure traps. The SNMP agent also captures PDU audit logs for Mirror-MIB operations. Configure the snmpTrap, snmpPduAudit, and snmpSetPduAudit logs at the proper severity level to capture the secure audit logs.

You can use the **show snmp secure-log** command to display the captured secure logs. Secure logs are stored in a string format similar to SNMP trap audit logs. You can use the **snmp-server clear secure-log** command to reset the secure logs.

The secure log configuration and data are not persistent. Secure audit logs are not available after a warm or cold restart of the SNMP agent, because the SNMP agent does not store the secure logs in NVS. The SNMP agent can store a maximum of 100 secure logs before overwriting the logs.

To enhance security, you can configure and display the secure audit logs only through the CLI. You cannot use SNMP to configure and display the logs. Secure trap logs are not populated in the notification logs MIB. From the perspective of the notification log MIB, secure traps do not exist.

Related Topics

- **snmp-server clear secure-log** command
- **snmp-server secure-log** command
- **show snmp secure-log** command
- **show snmp trap** command