Configuring Traffic Mirroring

This module describes the configuration of the traffic mirroring feature. Traffic mirroring is sometimes called port mirroring, or switched port analyzer (SPAN).

Feature History for Traffic Mirroring

Release 3.9.1	This feature was introduced.
Release 4.0.1	These traffic mirroring features were added: • Traffic mirroring over a pseudowire • Flow or ACL-based traffic mirroring • Layer 3 interface support • Partial packet mirroring
Release 5.1.0	The Sampled Traffic Mirroring feature was introduced.
Release 7.1.2	The SPAN to File feature was introduced.
Release 7.1.2	The File Mirroring feature was introduced.

Introduction to Traffic Mirroring

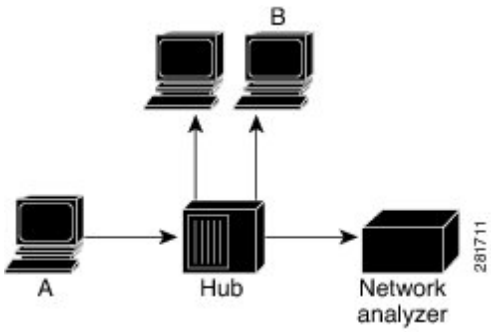
Traffic mirroring, which is sometimes called port mirroring, or Switched Port Analyzer (SPAN) is a Cisco proprietary feature that enables you to monitor Layer 2 or Layer 3 network traffic passing in, or out of, a set of Ethernet interfaces. You can then pass this traffic to a network analyzer for analysis.

Traffic mirroring copies traffic from one or more Layer 3 or Layer 2 interfaces or sub-interfaces, including Layer 2 link bundle interfaces or sub-interfaces, and sends the copied traffic to one or more destinations for analysis by a network analyzer or other monitoring device. Traffic mirroring does not affect the switching of traffic on the source interfaces or sub-interfaces, and allows the mirrored traffic to be sent to a destination interfaceor sub-interface.

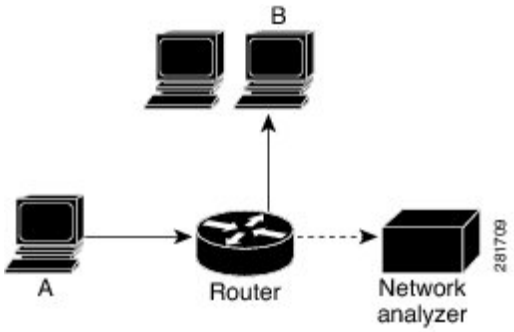
Traffic mirroring was introduced on switches because of a fundamental difference between switches and hubs. When a hub receives a packet on one port, the hub sends out a copy of that packet from all ports except from the one at which the hub received the packet. In the case of switches, after a switch boots, it starts to build up a Layer 2 forwarding table on the basis of the source MAC address of the different packets that the switch receives. After this forwarding table is built, the switch forwards traffic that is destined for a MAC address directly to the corresponding port.

For example, if you want to capture Ethernet traffic that is sent by host A to host B, and both are connected to a hub, just attach a traffic analyzer to this hub. All other ports see the traffic between hosts A and B.

Figure 1: Traffic Mirroring Operation on a Hub



On a switch or router, after the host B MAC address is learned, unicast traffic from A to B is only forwarded to the B port. Therefore, the traffic analyzer does not see this traffic.



In this configuration, the traffic analyzer captures only traffic that is flooded to all ports, such as:

- Broadcast traffic
- Multicast traffic with CGMP or Internet Group Management Protocol (IGMP) snooping disabled
- Unknown unicast traffic on a switch

An extra feature is necessary that artificially copies unicast packets that host A sends. This extra feature is traffic mirroring. When traffic mirroring is enabled, the traffic analyzer is attached to a port that is configured to receive a copy of every packet that host A sends. This port is called a traffic mirroring port. The other sections of this document describe how you can fine tune this feature.

Sampled Traffic Mirroring

SPAN is supported on all types of forwarding interfaces of the main interface (port level) such as, L2, L3 interfaces, sub-interface, bundle interface, and BNG interface. But Sampled SPAN is supported only at port level. Sampled SPAN is configurable in ingress direction only. SPAN and Sampled SPAN cannot be configured at the same time on main interface (at port level). When Sampled SPAN is enabled on main interface, SPAN is still configurable on rest of the forwarding interfaces on the port.

When Sampled SPAN is enabled on the underlying physical port and SPAN is configured on a forwarding interface, the packets are mirrored as follows:

- Sampled packet on the physical port is mirrored just to the destination port of the Sampled SPAN session.
- Non-sampled packet is mirrored for each of the regular SPAN session on the associated forwarding interface.

Sampled Traffic Mirroring allows you to:

1. Sample the packets based on a configured interval.
2. Apply Sampled SPAN on a physical port in order to include all forwarding interfaces on that port.
3. Configure the Sampling rate of monitoring that is configured for each source port. You can choose to configure one of these sampling rates; 1K, 2K, 4K, 8K, and 16K. For example, when 4K is configured as the sampling rate, for every 4K packets on the source port one packet is sampled and mirrored to the destination port.
4. Use Sampled SPAN along with Traffic Mirroring.
5. Enable Sampled SPAN on every bundle member, if the physical port is part of a link bundle.
6. Use all destination ports that were supported for SPAN.
7. Enable statistics support on each monitoring session.
8. Truncate and mirror a fixed portion of each mirrored packet (for example, the first 64 bytes of every packet received from the source port is mirrored to the destination port). You can configure the offset or the amount of fixed portion.

You can configure these source to destination combinations in sampled SPAN:

- Physical Port mirrored to Physical Port
- Physical Port mirrored to Pseudo-wire
- Bundle member port mirrored to Physical Port
- Bundle member port mirrored to Pseudo-wire

Implementing Traffic Mirroring on the Cisco ASR 9000 Series Router

Traffic Mirroring Terminology

- Ingress traffic—Traffic that enters the switch.
- Egress traffic—Traffic that leaves the switch.

- Source port—A port that is monitored with the use of traffic mirroring. It is also called a monitored port.
- Destination port—A port that monitors source ports, usually where a network analyzer is connected. It is also called a monitoring port.
- Monitor session—A designation for a collection of traffic mirroring configurations consisting of a single destination and, potentially, many source interfaces.

Characteristics of the Source Port

A source port, also called a monitored port, is a switched or routed port that you monitor for network traffic analysis. In a single local or remote traffic mirroring session, you can monitor source port traffic, such as received (Rx) for ingress traffic, transmitted (Tx) for egress traffic, or bidirectional (for both ingress and egress traffic). Your router can support any number of source ports (up to a maximum number of 800).

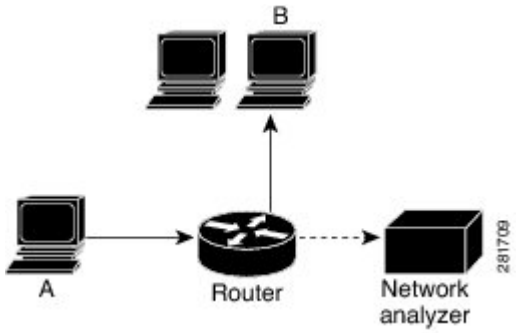
A source port has these characteristics:

- It can be any port type, such as Bundle Interface, Gigabit Ethernet, 10-Gigabit Ethernet, or EFPs.

Note Bridge group virtual interfaces (BVIs) are not supported.

- Each source port can be monitored in only one traffic mirroring session.
- It cannot be a destination port.
- Partial Packet Mirroring. The first 64 to 256 bytes of the packet can be mirrored.
- Each source port can be configured with a direction (ingress, egress, or both) to monitor. For bundles, the monitored direction applies to all physical ports in the group.

Figure 2: Network Analysis on a Cisco ASR 9000 RouterWith Traffic Mirroring



In the figure above, the network analyzer is attached to a port that is configured to receive a copy of every packet that host A sends. This port is called a traffic mirroring port.

Characteristics of the Monitor Session

A monitor session is a collection of traffic mirroring configurations consisting of a single destination and, potentially, many source interfaces. For any given monitor session, the traffic from the source interfaces (called *source ports*) is sent to the monitoring port (called the *destination port*) . Some optional operations such as VLAN tag imposition and ACL filtering can be performed on the mirrored traffic streams. If there is

more than one source port in a monitoring session, the traffic from the several mirrored traffic streams is combined at the destination port. The result is that the traffic that comes out of the destination port is a combination of the traffic from one or more source ports, and the traffic from each source port may or may not have VLAN push operations or ACLs applied to it.

Monitor sessions have these characteristics:

- A single Cisco ASR 9000 Router can have a maximum of eight monitor sessions.
- A single monitor session can have only one destination port.
- A single destination port can belong to only one monitor session.
- A single Cisco ASR 9000 Router can have a maximum of 800 source ports.
- A monitor session can have a maximum of 800 source ports, as long as the maximum number of source ports from all monitoring sessions does not exceed 800.

Characteristics of the Destination Port

Table 1: Feature History Table

Feature Name	Release	Description
SPAN support for bundle as a destination interface	Release 7.5.1	This feature adds the ability to configure a bundle destination for SPAN. Bundle interfaces increase the bandwidth by allowing forwarding across all available members of the bundle. Additionally, they provide redundancy in the case of link failure.

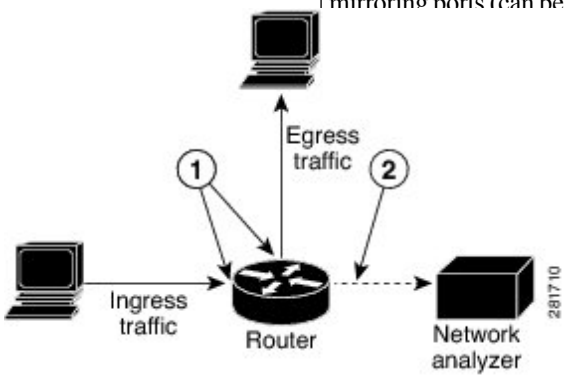
Each local session or remote destination session must have a destination port (also called a monitoring port) that receives a copy of the traffic from the source ports.

A destination port has these characteristics:

- A destination port must reside on the same router as the source port.
- A destination port can be any Ethernet physical port, EFP, pseudowire, or a bundle interface.
- A destination port can only be a Layer 2 transport interface. An L3 interface as a SPAN destination cannot be configured on the Cisco ASR 9000 Series Router.
- A destination port can be a trunk (main) interface or a subinterface.
- At any one time, a destination port can participate in only one traffic mirroring session. A destination port in one traffic mirroring session cannot be a destination port for a second traffic mirroring session. In other words, no two monitor sessions can have the same destination port.
- A destination port cannot also be a source port.

- *Figure 3: Network Analysis on a Cisco ASR 9000 Router With Traffic Mirroring*

1	Source traffic mirroring ports (can be	2	Destination traffic mirroring port



Supported Traffic Mirroring Types

These traffic mirroring types are supported:

- Local traffic mirroring. This is the most basic form of traffic mirroring. The network analyzer or sniffer is directly attached to the destination interface. In other words, all monitored ports are all located on the same switch as the destination port.
- Remote traffic mirroring (known as R-SPAN). In this case, the network analyzer is not attached directly to the destination interface, but is on a VLAN accessible to the switch. For example, the destination interface is a sub-interface with a VLAN encapsulation.

  A restricted form of remote traffic mirroring can be implemented by sending traffic to a single destination port that pushes a VLAN tag, instead of switching through a bridge domain. Remote traffic mirroring:

  - Allows decoupling of the network analyzer and destination, but there is no on-the-box redundancy.
  - Allows multiple remote network analyzers as long as they can attach to the traffic mirroring VLAN.

    This is supported on Cisco IOS XR software because the destination port is an EFP that can push a VLAN tag.

- Pseudowire traffic mirroring (known as PW-SPAN in Cisco IOS Software). Instead of using a standard destination interface, traffic is mirrored to a remote site through an MPLS pseudowire.
- ACL-based traffic mirroring. Traffic is mirrored based on the configuration of the global interface ACL.
- Partial Packet Mirroring. The first 64 to 256 bytes of the packet can be mirrored.
- Layer 2 or Layer 3 traffic mirroring is supported. Both Layer 2 and Layer 3 source ports can be mirrored.

Pseudowire Traffic Mirroring

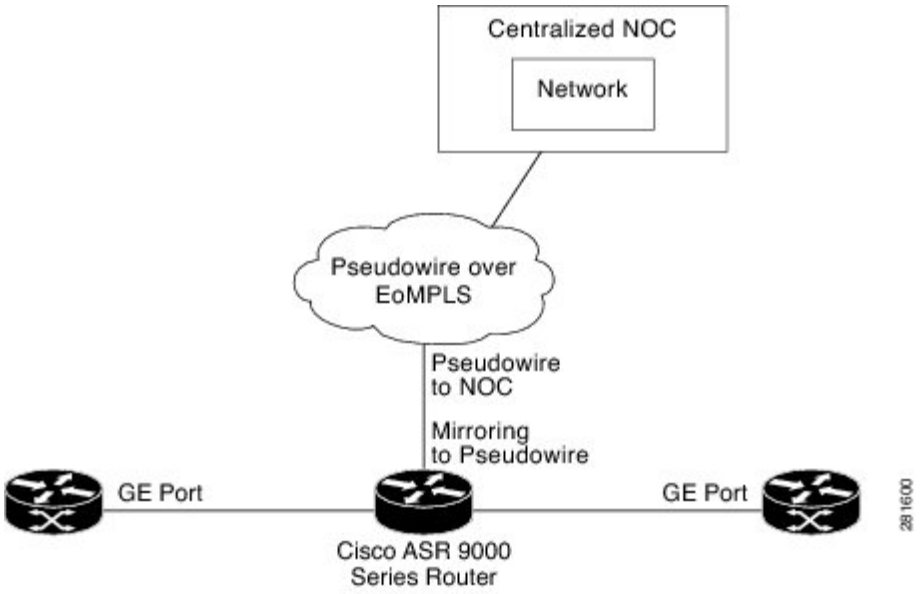
You can configure the traffic mirroring destination port to be a pseudowire rather than a physical port. In this case, the system mirrors the designated traffic on the source port over the pseudowire to a central location. This allows the centralization of expensive network traffic analysis tools.

Because the pseudowire carries only mirrored traffic, this traffic is unidirectional. There must not be any traffic coming from the remote provider edge.

In such a pseudowire traffic mirroring scenario, though the system mirrors traffic successfully, the statistics for sent pseudowire packet statistics remains zero.

To protect the pseudowire traffic mirroring path against network failures, it is possible to configure a traffic engineering tunnel as the preferred path and enable fast reroute protection for the pseudowire.

Figure 4: Pseudowire Traffic Mirroring



ACL-Based Traffic Mirroring

You can mirror traffic based on the definition of a global interface access list (ACL). If you are mirroring Layer 2 traffic, the ACL is configured using the **ethernet-services access-list** command with the **capture** keyword. When you are mirroring Layer 3 traffic, the ACL is configured using the **ipv4 access-list** or **ipv6 access-list** command with the **capture** keyword. The **permit** and **deny** commands determine the behavior of regular traffic. The **capture** keyword designates that the packet is to be mirrored to the destination port.

Restrictions for Traffic Mirroring

A maximum of eight monitoring sessions are supported. You can configure 800 source ports on a single monitoring session or an aggregate of 800 source ports over eight monitoring sessions.

These forms of traffic mirroring are not supported:

- ACL-Based SPAN on Layer 3 interface configured with a dot1q encapsulation.
- Mirroring traffic to a GRE tunnel (also known as Encapsulated Remote Switched Port Analyzer [ER-SPAN] in Cisco IOS Software).
- Mirroring traffic from a full bridge domain (also known as VLAN-based SPAN in Cisco IOS Software).

- Mirroring traffic on an individual bundle member interface is not supported. SPAN must be configured only on a bundle interface and it is applied to all members.
- If the destination of traffic mirroring is an nV satellite port and ICL is configured with a bundle interface, then replicated packets are not forwarded to the destination.
- The system does not support MAP-T inline and SPAN on the same NPU.
- To avoid traffic loss, disable SPAN, if enabled on MAP-E/T service-inline interfaces.
- SPAN is not supported on those line card ports that are carrying traffic bound for a VSM. This behaviour is observed only on the Cisco ASR 9000 High Density 100GE Ethernet line cards and Cisco ASR 9000 Series 24-Port and 48-Port Dual-Rate 10GE/1GE line cards.
- When you configure ingress port SPAN on an interface, BFD sessions such as BFD-over-BVI, may encounter flaps during traffic congestion. This happens because the BFD-over-BVI traffic is handled via Priority Normal Traffic Manager Loopback queue in spite of prioritising the BFD-over-BVI packets. Except for the Cisco ASR 9000 Series 5th Generation High-Density Line Cards and Cisco ASR 9000 Series 4th Generation QSFP28 based dense 100GE Line Cards, this limitation is observed in all the other line cards.
- On Cisco ASR 9903 routers, the Online Insertion and Removal (OIR) of a Port Expansion Card (PEC) with BFD sessions, support 300ms asynchronous timers and 150ms echo timers. BFD sessions with less than the supported timer values may encounter flaps during the PEC OIR.

Restrictions of Sampled Traffic Mirroring

These are the restrictions of Sampled Traffic Mirroring:

- Sampled SPAN can be applied to ingress traffic only.
- The source for sampled SPAN must be on Cisco ASR 9000 Enhanced Ethernet Line Cards.
- Sampled SPAN works only on physical interfaces.
- The source port cannot be on bundles; however it can be applied to bundle member links.
- Sampled SPAN does not work on sub-interfaces, however it can be applied to a physical port with sub-interfaces(main port).
- Only these intervals are accepted: 512, 1K, 2K, 4K, 8K, and 16K. The default interval is 16K.
- Sampled SPAN is configurable at physical port level only.
- Sampled SPAN rate is ingress port specific and not session specific. This means that a destination port can take multiple ingress sampled ports at different sampling rates.
- In the case of a bundle interface, you must configure Sampled SPAN on all the physical ports that are members of the bundle.
- ACL filtering is not supported for Sampled Mirrored Traffic.

Performance Impact with Traffic Mirroring

It is recommended that you do not mirror more than 15% of your total transit traffic. On the Cisco ASR 9000 Ethernet Line Card, that uses Ten Gigabit Ethernet interfaces or bundle interfaces there is a limit of 1.5G of data on each of the ingress and egress traffic that can be mirrored. This limitation is not applicable on the Cisco ASR 9000 Enhanced Ethernet Line Card.

Configuring Traffic Mirroring

These tasks describe how to configure traffic mirroring:

How to Configure Local Traffic Mirroring

SUMMARY STEPS

1. **configure**
2. **monitor-session** *session-name*
3. **destination interface** *dest-interface*
4. **exit**
5. **interface** *source-interface*
6. **l2transport**
7. **monitor-session** *session-name* [**direction** {**rx-only** | **tx-only**]
8. **end** or **commit**
9. **show monitor-session [session-name] status** [**detail**] [**error**]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure** **Example:** `RP/0/RSP0/CPU0:router# configure`	Enters global configuration mode.

Step 2	**monitor-session** *session-name*
	Example:
	`RP/0/RSP0/CPU0:router(config)# monitor-session mon1`
	`RP/0/RSP0/CPU0:router(config-mon)#`
		Defines a monitor session and enters monitor session configuration mode.

Step 3	**destination interface** *dest-interface* **Example:** `RP/0/RSP0/CPU0:router(config-mon)# destination interface gigabitethernet0/0/0/15`	Specifies the destination interface to which traffic is replicated.

	Command or Action	Purpose

Step 4	**exit**
	Example:
	`RP/0/RSP0/CPU0:router(config-mon)# exit`
	`RP/0/RSP0/CPU0:router(config)#`
		Exits monitor session configuration mode and returns to global configuration mode.

Step 5	**interface** *source-interface* **Example:** `RP/0/RSP0/CPU0:router(config)# interface gigabitethernet0/0/0/11`	Enters interface configuration mode for the specified interface. The interface number is entered in *rack*/*slot*/*module*/*port* notation. For more information about the syntax for the router, use the question mark (?) online help function.
Step 6	**l2transport** **Example:** `RP/0/RSP0/CPU0:router(config-if)# l2transport`	(Optional) Enables Layer 2 transport mode on the interface and enters Layer 2 transport configuration mode. **Note** • Use the **l2transport** command to mirror all traffic types.
Step 7	**monitor-session** *session-name* [**direction** {**rx-only** | **tx-only**] **Example:** `RP/0/RSP0/CPU0:router(config-if-l2)# monitor-session mon1`	Specifies the monitor session to be used on this interface. Use the **direction** keyword to specify that only ingress or only egress traffic is mirrored.

Step 8	**end** or **commit**
	Example:
	`RP/0/RSP0/CPU0:router(config-if-l2)# end`
	or
	`RP/0/RSP0/CPU0:router(config-if-l2)# commit`
		Saves configuration changes.
		• When you issue the **end** command, the system prompts you to commit changes:
		`Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]:`
		- Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
		- Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
		- Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes.
		• Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.

	Command or Action	Purpose
Step 9	**show monitor-session [session-name] status** [**detail**] [**error**] **Example:** `RP/0/RSP0/CPU0:router# show monitor-session`	Displays information about the monitor session.

How to Configure Remote Traffic Mirroring

SUMMARY STEPS

1. **configure**
2. **monitor-session** *session-name*
3. **destination interface** *dest-subinterface*
4. **exit**
5. **interface** *dest-subinterface* **l2transport**
6. **encapsulation dot1q** *vlan*
7. **rewrite ingress tag pop** *tag-to-remove*
8. **interface** *source-subinterface* [**l2transport**]
9. **monitor-session** *session-name* [**direction** {**rx-only** | **tx-only**]
10. **end** or **commit**
11. **show monitor-session [session-name] status** [**detail**] [**error**]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure** **Example:** `RP/0/RSP0/CPU0:router# configure`	Enters global configuration mode.

Step 2	**monitor-session** *session-name*
	Example:
	`RP/0/RSP0/CPU0:router(config)# monitor-session mon1`
	`RP/0/RSP0/CPU0:router(config-mon)#`
		Defines a monitor session and enters monitor session configuration mode.

Step 3	**destination interface** *dest-subinterface* **Example:** `RP/0/RSP0/CPU0:router(config-mon)# destination interface gigabitethernet0/0/0/15`	Specifies the destination subinterface to which traffic is replicated.
Step 4	**exit** **Example:**	Exits monitor session configuration mode and returns to global configuration mode.

	Command or Action	Purpose
	`RP/0/RSP0/CPU0:router(config-mon)# exit` `RP/0/RSP0/CPU0:router(config)#`	
Step 5	**interface** *dest-subinterface* **l2transport** **Example:** `RP/0/RSP0/CPU0:router(config)# interface gigabitethernet0/0/0/11.10 l2transport`	Enters interface configuration mode for the specified sub-interface. The interface number is entered in *rack*/*slot*/*module*/*port* notation. For more information about the syntax for the router, use the question mark (?) online help function. The **l2transport** keyword is used to enable Layer 2 transport mode on the destination subinterface.
Step 6	**encapsulation dot1q** *vlan* **Example:** `RP/0/RSP0/CPU0:router(config-if)# encapsulation dot1q 1`	Specifies 802.1Q encapsulation and the VLAN number that is used.
Step 7	**rewrite ingress tag pop** *tag-to-remove* **Example:** `RP/0/RSP0/CPU0:router(config-if)# rewrite ingress tag pop 1`	Specifies to remove the outer tag only for the EFP.
Step 8	**interface** *source-subinterface* [**l2transport**] **Example:** `RP/0/RSP0/CPU0:router(config)# interface gigabitethernet0/0/0/11.10 l2transport`	Enters interface configuration mode for the specified subinterface. The interface number is entered in *rack*/*slot*/*module*/*port* notation. For more information about the syntax for the router, use the question mark (?) online help function. To configure a Layer 2 subinterface to be the source interface, use the **l2transport** keyword to enable Layer 2 transport mode on the subinterface.
Step 9	**monitor-session** *session-name* [**direction** {**rx-only** | **tx-only**] **Example:** `RP/0/RSP0/CPU0:router(config-if-l2)# monitor-session mon1`	Specifies the monitor session to be used on this interface. Use the **direction** keyword to specify that only ingress or egress traffic is mirrored.

Step 10	**end** or **commit**
	Example:
	`RP/0/RSP0/CPU0:router(config-if)# end`
	or
	`RP/0/RSP0/CPU0:router(config-if)# commit`
		Saves configuration changes.
		• When you issue the **end** command, the system prompts you to commit changes:
		`Uncommitted changes found, commit them before exiting (yes/no/cancel)?`
		`[cancel]:`

	Command or Action	Purpose
		- Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode. - Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes. - Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes. • Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session. Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.
Step 11	**show monitor-session [session-name] status** [**detail**] [**error**] **Example:** `RP/0/RSP0/CPU0:router# show monitor-session`	Displays information about the traffic mirroring session.

How to Configure Traffic Mirroring over Pseudowire

SUMMARY STEPS

1. **configure**
2. **monitor-session** *session-name*
3. **destination psuedowire**
4. **exit**
5. **interface** *source-interface*
6. **l2transport**
7. **monitor-session** *session-name* [**direction** {**rx-only** | **tx-only**}]
8. **exit**
9. **exit**
10. **exit**
11. **l2vpn**
12. **pw-class** *class-name*
13. **encapsulation mpls**
14. **exit**
15. **exit**
16. **xconnect group** *group-name*
17. **p2p** *xconnect-name*

18. **monitor-session** *session-name*
19. **neighbor** *peer-ip* **pw-id** *pseudowire-id*
20. **pw-class** *class-name*
21. **end** or **commit**
22. **show monitor-session [session-name] status** [**detail**] [**error**]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure** **Example:** `RP/0/RSP0/CPU0:router# configure`	Enters global configuration mode.

Step 2	**monitor-session** *session-name*
	Example:
	`RP/0/RSP0/CPU0:router(config)# monitor-session mon1`
	`RP/0/RSP0/CPU0:router(config-mon)#`
		Defines a monitor session and enters monitor session configuration mode.

Step 3	**destination psuedowire** **Example:** `RP/0/RSP0/CPU0:router(config-mon)# destination pseudowire`	Specifies that the traffic is replicated to a pseudowire.

Step 4	**exit**
	Example:
	`RP/0/RSP0/CPU0:router(config-mon)# exit`
	`RP/0/RSP0/CPU0:router(config)#`
		Exits monitor session configuration mode and returns to global configuration mode.

Step 5	**interface** *source-interface* **Example:** `RP/0/RSP0/CPU0:router(config)# interface gigabitethernet0/0/0/11.10`	Enters interface configuration mode for the specified interface. The interface number is entered in *rack*/*slot*/*module*/*port* notation. For more information about the syntax for the router, use the question mark (?) online help function.
Step 6	**l2transport** **Example:** `RP/0/RSP0/CPU0:router(config-if)# l2transport`	(Optional) Enables Layer 2 transport mode on the subinterface and enters Layer 2 transport configuration mode. **Note** • Use the **l2transport** command to mirror all traffic types.
Step 7	**monitor-session** *session-name* [**direction** {**rx-only** | **tx-only**}] **Example:**	Specifies the monitor session to be used on this interface. Use the **direction** keyword to specify that only ingress or egress traffic is mirrored.

	Command or Action	Purpose
	`RP/0/RSP0/CPU0:router(config-if-l2)# monitor-session mon1`	

Step 8	**exit**
	Example:
	`RP/0/RSP0/CPU0:router(config-if-mon)# exit`
	`RP/0/RSP0/CPU0:router(config-if-l2)#`
		Exits monitor session configuration mode and returns to l2transport configuration mode.

Step 9	**exit**
	Example:
	`RP/0/RSP0/CPU0:router(config-if-l2)# exit`
	`RP/0/RSP0/CPU0:router(config-if)#`
		Exits l2transport configuration mode and returns to interface configuration mode.

Step 10	**exit**
	Example:
	`RP/0/RSP0/CPU0:router(config-if)# exit`
	`RP/0/RSP0/CPU0:router(config)#`
		Exits interface configuration mode and returns to global configuration mode.

Step 11	**l2vpn**
	Example:
	`RP/0/RSP0/CPU0:router(config)# l2vpn`
	`RP/0/RSP0/CPU0:router(config-l2vpn)#`
		Enters Layer 2 VPN configuration mode.

Step 12	**pw-class** *class-name* **Example:** `RP/0/RSP0/CPU0:router(config-l2vpn)# pw-class pw1`	Configures a pseudowire class template and enters pseudowire class template configuration mode.
Step 13	**encapsulation mpls** **Example:** `RP/0/RSP0/CPU0:router(config-l2vpn-pwc)# encapsulation mpls`	Configures the pseudowire encapsulation to MPLS.

Step 14	**exit**
	Example:
	`RP/0/RSP0/CPU0:router(config-l2vpn-pwc-mpls)# exit`
	`RP/0/RSP0/CPU0:router(config-l2vpn-pwc)`
		Exits pseudowire encapsulation configuration mode.

Step 15	**exit** **Example:**	Exits pseudowire class template configuration mode.

	Command or Action	Purpose
	`RP/0/RSP0/CPU0:router(config-l2vpn-pwc)# exit` `RP/0/RSP0/CPU0:router(config-l2vpn)`	
Step 16	**xconnect group** *group-name* **Example:** `RP/0/RSP0/CPU0:router(config-l2vpn)# xconnect group g1`	Configures a group cross connect.
Step 17	**p2p** *xconnect-name* **Example:** `RP/0/RSP0/CPU0:router(config-l2vpn-xc)# p2p xc1`	Configures a point-to-point cross connect.
Step 18	**monitor-session** *session-name* **Example:** `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# monitor-session mon1`	Attaches a traffic mirroring session to the point-to-point cross connect.
Step 19	**neighbor** *peer-ip* **pw-id** *pseudowire-id* **Example:** `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# neighbor 192.168.2.2 pw-id 3`	Configures the point-to-point cross connect.
Step 20	**pw-class** *class-name* **Example:** `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# pw-class pw1`	Specifies the pseudowire class template to use for the cross connect.

Step 21	**end** or **commit**
	Example:
	`RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p-pw)# end`
	or
	`RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p-pw)# commit`
		Saves configuration changes.
		• When you issue the **end** command, the system prompts you to commit changes:
		`Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]:`
		- Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
		- Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes.

	Command or Action	Purpose
		- Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes. • Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session. Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.
Step 22	**show monitor-session [session-name] status** [**detail**] [**error**] **Example:** `RP/0/RSP0/CPU0:router# show monitor-session`	Displays information about the traffic mirroring session.

How to Configure ACL-Based Traffic Mirroring

Before you begin

The global interface ACL should be configured using one of these commands with the **capture** keyword:

- **ipv4 access-list**
- **ipv6 access-list**
- **ethernet-services access-list**

  For more information, refer to the *Cisco ASR 9000 Series Aggregation Services Router IP Addresses and Services Command Reference* or the *ASR 9000 Series Aggregation Services Router L2 VPN and Ethernet Services Command Reference*.

SUMMARY STEPS

1. **configure**
2. **monitor-session** *session-name*
3. **destination interface** *dest-interface*
4. **exit**
5. **interface** *source-interface*
6. **l2transport**
7. **exit**
8. **ethernet-services access-group** *access-list-name* [**ingress** | **egress**]
9. **monitor-session** *session-name* [ipv4|ipv6] [direction {rx-only|tx-only}]
10. **acl**
11. **end** or **commit**
12. **show monitor-session [session-name] status** [**detail**] [**error**]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure** **Example:** `RP/0/RSP0/CPU0:router# configure`	Enters global configuration mode.

Step 2	**monitor-session** *session-name*
	Example:
	`RP/0/RSP0/CPU0:router(config)# monitor-session mon1`
	`RP/0/RSP0/CPU0:router(config-mon)#`
		Defines a monitor session and enters monitor session configuration mode.

Step 3	**destination interface** *dest-interface* **Example:** `RP/0/RSP0/CPU0:router(config-mon)# destination interface gigabitethernet0/0/0/15`	Specifies the destination interface to which traffic should be replicated.

Step 4	**exit**
	Example:
	`RP/0/RSP0/CPU0:router(config-mon)# exit`
	`RP/0/RSP0/CPU0:router(config)#`
		Exits monitor session configuration mode and returns to global configuration mode.

Step 5	**interface** *source-interface* **Example:** `RP/0/RSP0/CPU0:router(config)# interface gigabitethernet0/0/0/11`	Enters interface configuration mode for the specified interface. The interface number is entered in *rack*/*slot*/*module*/*port* notation. For more information about the syntax for the router, use the question mark (?) online help function.
Step 6	**l2transport** **Example:** `RP/0/RSP0/CPU0:router(config-if)# l2transport`	(Optional) Enables Layer 2 transport mode on the subinterface and enters Layer 2 transport configuration mode. **Note** • Use the **l2transport** command to mirror all traffic types.

Step 7	**exit**
	Example:
	`RP/0/RSP0/CPU0:router(config-if-l2)# exit`
	`RP/0/RSP0/CPU0:router(config-if)#`
		Exits Layer 2 transport configuration mode and returns to interface configuration mode.

Step 8	**ethernet-services access-group** *access-list-name* [**ingress** | **egress**] **Example:**	Associates the access list definition with the interface being mirrored.

	Command or Action	Purpose
	`RP/0/RSP0/CPU0:router(config-if)# ethernet-services access-group acl1 ingress`	
Step 9	**monitor-session** *session-name* [ipv4|ipv6] [direction {rx-only|tx-only}] **Example:** `RP/0/RSP0/CPU0:router(config-if)# monitor-session mon1 direction rx-only`	Specifies the monitor session to be used on this interface.
Step 10	**acl** **Example:** `RP/0/RSP0/CPU0:router(config-if-mon)# acl`	Specifies that the traffic mirrored is according to the defined global interface ACL.

Step 11	**end** or **commit**
	Example:
	`RP/0/RSP0/CPU0:router(config-if)# end`
	or
	`RP/0/RSP0/CPU0:router(config-if)# commit`
		Saves configuration changes.
		• When you issue the **end** command, the system prompts you to commit changes:
		`Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]:`
		- Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
		- Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
		- Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes.
		• Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.

Step 12	**show monitor-session [session-name] status** [**detail**] [**error**] **Example:** `RP/0/RSP0/CPU0:router# show monitor-session`	Displays information about the monitor session.

Troubleshooting ACL-Based Traffic Mirroring

Take note of these configuration issues:

- If an ACL-Based SPAN is configured on a Layer 3 interface configured with a dot1q encapsulation, the configuration will not commit.
- Even when the **acl** command is configured on the source mirroring port, if the ACL configuration command does not use the **capture** keyword, no traffic gets mirrored.
- If the ACL configuration uses the **capture** keyword, but the **acl** command is not configured on the source port, traffic is mirrored, but no access list configuration is applied.
- All ingress traffic is mirrored, regardless of the ACL definition; only egress traffic permitted in the ACL definition is mirrored.

This example shows both the **capture** keyword in the ACL definition and the **acl** command configured on the interface:

```
monitor-session tm_example
!
ethernet-services access-list tm_filter
 10 deny 0000.1234.5678 0000.abcd.abcd any capture
!
interface GigabitEthernet0/2/0/0
 monitor-session tm_example direction rx-only
 acl
!
 l2transport
 !
 ethernet-services access-group tm_filter ingress
end
```

How to Configure Partial Packet Mirroring

SUMMARY STEPS

1. **configure**
2. **monitor-session** *session-name*
3. **destination interface** *dest-interface*
4. **exit**
5. **interface** *source-interface*
6. **monitor-session** *session-name*[**direction** {**rx-only** | **tx-only**]
7. **mirror first** *bytes*
8. **end** or **commit**
9. **show monitor-session [session-name] status**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure** **Example:** `RP/0/RSP0/CPU0:router# configure`	Enters global configuration mode.

	Command or Action	Purpose

Step 2	**monitor-session** *session-name*
	Example:
	`RP/0/RSP0/CPU0:router(config)# monitor-session mon1`
	`RP/0/RSP0/CPU0:router(config-mon)#`
		Defines a monitor session and enters monitor session configuration mode.

Step 3	**destination interface** *dest-interface* **Example:** `RP/0/RSP0/CPU0:router(config-mon)# destination interface gigabitethernet0/0/0/15`	Specifies the destination interface to which traffic should be replicated.

Step 4	**exit**
	Example:
	`RP/0/RSP0/CPU0:router(config-mon)# exit`
	`RP/0/RSP0/CPU0:router(config)#`
		Exits monitor session configuration mode and returns to global configuration mode.

Step 5	**interface** *source-interface* **Example:** `RP/0/RSP0/CPU0:router(config)# interface gigabitethernet0/0/0/11.10`	Enters interface configuration mode for the specified interface. The interface number is entered in *rack*/*slot*/*module*/*port* notation. For more information about the syntax for the router, use the question mark (?) online help function.
Step 6	**monitor-session** *session-name*[**direction** {**rx-only** | **tx-only**] **Example:** `RP/0/RSP0/CPU0:router(config-if-l2)# monitor-session mon1`	Specifies the monitor session to be used on this interface. Use the **direction** keyword to specify that only ingress or egress traffic is mirrored.
Step 7	**mirror first** *bytes* **Example:** `RP/0/RSP0/CPU0:router(config-if-mon)# mirror first bytes`	Specifies the number of bytes of the packet to mirror. Values can range from 64 to 256.

Step 8	**end** or **commit**
	Example:
	`RP/0/RSP0/CPU0:router(config-if)# end`
	or
	`RP/0/RSP0/CPU0:router(config-if)# commit`
		Saves configuration changes.
		• When you issue the **end** command, the system prompts you to commit changes:
		`Uncommitted changes found, commit them before exiting (yes/no/cancel)?`
		`[cancel]:`
		- Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.

	Command or Action	Purpose
		- Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes. - Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes. • Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.
Step 9	**show monitor-session [session-name] status** **Example:** `RP/0/RSP0/CPU0:router# show monitor-session`	Displays information about the traffic mirroring session.

Traffic Mirroring Configuration Examples

This section contains examples of how to configure traffic mirroring:

Traffic Mirroring with Physical Interfaces (Local): Example

This example shows a basic configuration for traffic mirroring with physical interfaces. When traffic flows over the point-to-point cross connect between gig0/2/0/19 and gig0/2/0/11, packets received and transmitted on gig0/2/0/19 are also mirrored to gig0/2/0/15.

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# monitor-session ms1
RP/0/RSP0/CPU0:router(config-mon)# destination interface gig0/2/0/15
RP/0/RSP0/CPU0:router(config-mon)# commit

RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# interface gig0/2/0/11
RP/0/RSP0/CPU0:router(config-subif)# l2transport
RP/0/RSP0/CPU0:router(config-if-l2)# commit

RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# interface gig0/2/0/15
RP/0/RSP0/CPU0:router(config-subif)# l2transport
RP/0/RSP0/CPU0:router(config-if-l2)# commit

RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# interface gig0/2/0/19
RP/0/RSP0/CPU0:router(config-subif)# l2transport
RP/0/RSP0/CPU0:router(config-subif-l2)# monitor-session ms1
RP/0/RSP0/CPU0:router(config-if-l2)# commit

RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# l2vpn
RP/0/RSP0/CPU0:router(config-l2vpn)# xconnect group xg1
```

```
RP/0/RSP0/CPU0:router(config-l2vpn-xc)# p2p xg1_p1
RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# interface gig0/2/0/11
RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# interface gig0/2/0/19
RP/0/RSP0/CPU0:router(config-if-l2)# commit
```

Traffic Mirroring with EFPs (Remote): Example

This example shows a basic configuration for remote traffic mirroring with EFP interfaces. When traffic flows over the point-to-point cross connect between gig0/2/0/19.10 and gig0/2/0/11.10, packets received and transmitted on gig0/2/0/19.10 are also mirrored to gig0/2/0/10.1.

```
RP/0/RSP0/CPU0:router#monitor-session ms1
RP/0/RSP0/CPU0:router(config)# destination interface gig0/2/0/10.1

RP/0/RSP0/CPU0:router(config)# interface gig0/2/0/10.1 l2transport
RP/0/RSP0/CPU0:router(config-if-l2)# encapsulation dot1q 1
RP/0/RSP0/CPU0:router(config-if-l2)# rewrite ingress tag pop 1

RP/0/RSP0/CPU0:router(config)# interface gig0/2/0/11.10 l2transport
RP/0/RSP0/CPU0:router(config-if-l2)# encapsulation dot1q 10

RP/0/RSP0/CPU0:router(config)# interface gig0/2/0/19.10 l2transport
RP/0/RSP0/CPU0:router(config-if-l2)# encapsulation dot1q 10
RP/0/RSP0/CPU0:router(config-if-l2)# monitor-session ms1

RP/0/RSP0/CPU0:router(config)# l2vpn
RP/0/RSP0/CPU0:router(config-l2vpn)# xconnect group xg1
RP/0/RSP0/CPU0:router(config-l2vpn-xc)# p2p xg1_p1
RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# interface gig0/2/0/11.10
RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# interface gig0/2/0/19.10
```

Traffic Mirroring with Bundle Interfaces: Example

This example shows a basic configuration for traffic monitoring with bundle interfaces.

```
Router# configure
Router(config)# monitor-session ms1
Router(config-mon)# destination interface bundle-ether1
Router(config-mon)# commit

Router# configure
Router(config)# interface bundle-ether1
Router(config-if)# l2transport
Router(config-if-l2)# commit
```

Viewing Monitor Session Status: Example

This example shows sample output of the **show monitor-session** command with the **status** keyword:

```
RP/0/RSP0/CPU0:router# show monitor-session status

Monitor-session cisco-rtp1
Destination interface GigabitEthernet0/5/0/38
================================================================================
Source Interface   Dir  Status
---------------------- ---- --------------------------------------------------
```

```
Gi0/5/0/4       Both Operational
Gi0/5/0/17      Both Operational


Router# show monitor-session status

Monitor-session cisco-rtp1
Destination interface bundle-ether1
================================================================================
Source Interface   Dir  Status
---------------------- ------ -----------------------------------
Gi0/5/0/4       Both Operational
Gi0/5/0/17      Both Operational


RP/0/RSP0/CPU0:router# show monitor-session status detail

Monitor-session sess1
 Destination interface is not configured
 Source Interfaces
 -----------------
 GigabitEthernet0/0/0/0
  Direction: Both
  ACL match: Enabled
  Portion:  Full packet
  Status:  Not operational (destination interface not known).
 GigabitEthernet0/0/0/2
  Direction: Both
  ACL match: Disabled
  Portion:  First 100 bytes


RP/0/RSP0/CPU0:router# show monitor-session status error

Monitor-session ms1
Destination interface GigabitEthernet0/2/0/15 is not configured
================================================================================
Source Interface   Dir  Status
---------------------- ---- -----------------------------------------------------

Monitor-session ms2
Destination interface is not configured
================================================================================
Source Interface   Dir  Status
---------------------- ---- -----------------------------------------------------
```

Monitor Session Statistics: Example

Use the **show monitor-session** command with the **counters** keyword to show the statistics/counters (received/transmitted/dropped) of different source ports. For each monitor session, this command displays a list of all source interfaces and the replicated packet statistics for that interface.

The full set of statistics displayed for each interface is:

- RX replicated packets and octets
- TX replicated packets and octets
- Non-replicated packet and octets

```
RP/0/RSP0/CPU0:router# show monitor-session counters

Monitor-session ms1
 GigabitEthernet0/2/0/19.10
  Rx replicated: 1000 packets, 68000 octets
  Tx replicated: 1000 packets, 68000 octets
  Non-replicated: 0 packets, 0 octets
```

Use the **clear monitor-session counters** command to clear any collected statistics. By default this command clears all stored statistics; however, an optional interface filter can be supplied.

```
RP/0/RSP0/CPU0:router# clear monitor-session counters
```

Traffic Mirroring over Pseudowire: Example

This example shows how to configure traffic mirroring over a pseudowire:

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# interface GigabitEthernet0/11/0/1
RP/0/RSP0/CPU0:router(config-if)# l2transport
RP/0/RSP0/CPU0:router(config-if-l2)# monitor-session pw-span-test

RP/0/RSP0/CPU0:router(config)# monitor-session pw-span-test
RP/0/RSP0/CPU0:router(config-mon)# destination pseudowire

RP/0/RSP0/CPU0:router(config)# l2vpn
RP/0/RSP0/CPU0:router(config-l2vpn)# pw-class class1
RP/0/RSP0/CPU0:router(config-l2vpn-pwc)# encapsulation mpls

RP/0/RSP0/CPU0:router(config-l2vpn)# xconnect group g1
RP/0/RSP0/CPU0:router(config-l2vpn-xc)# p2p x1
RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# monitor-session pw-span-test
RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# neighbor 2.2.2.2 pw-id 1
RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p-pw)# pw-class class1

RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p-pw)# commit
```

Layer 3 ACL-Based Traffic Mirroring: Example

This example shows how to configure Layer 3 ACL-based traffic mirroring:

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# monitor-session ms1
RP/0/RSP0/CPU0:router(config-mon)# destination
RP/0/RSP0/CPU0:router(config-mon)# commit

RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# interface gig0/2/0/11
RP/0/RSP0/CPU0:router(config-if)# ipv4 access-group span ingress
RP/0/RSP0/CPU0:router(config-if)# monitor-session ms1
RP/0/RSP0/CPU0:router(config-if-mon)# commit

RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# ipv4 access-list span
RP/0/RSP0/CPU0:router(config-ipv4-acl)# 5 permit ipv4 any any dscp 5 capture
```

```
RP/0/RSP0/CPU0:router(config-ipv4-acl)# 10 permit ipv4 any any
RP/0/RSP0/CPU0:router(config-ipv4-acl)# commit
```

Layer 2 ACL-Based Traffic Mirroring: Example

This example shows how to configure Layer 2 ACL-based traffic mirroring:

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# monitor-session ms1
RP/0/RSP0/CPU0:router(config-mon)# destination interface gig0/2/0/15
RP/0/RSP0/CPU0:router(config-mon)# commit

RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# interface gig0/2/0/11
RP/0/RSP0/CPU0:router(config-if)# l2transport
RP/0/RSP0/CPU0:router(config-if-l2)# exit
RP/0/RSP0/CPU0:router(config-if)# ethernet-services access-group acl_mirror ingress
RP/0/RSP0/CPU0:router(config-if)# acl
RP/0/RSP0/CPU0:router(config-if)# monitor-session ms1
RP/0/RSP0/CPU0:router(config-if-mon)# commit

RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# ipv4 access-list acl_mirror
RP/0/RSP0/CPU0:router(config-ipv4-acl)# 5 permit ipv4 any any dscp 5 capture
RP/0/RSP0/CPU0:router(config-ipv4-acl)# 10 permit ipv4 any any
RP/0/RSP0/CPU0:router(config-ipv4-acl)# commit
```

Partial Packet Mirroring: Example

This example shows how to configure mirroring of the first 100 bytes of the packet:

```
RP/0/RP0/CPU0:router(config)# interface gigabitethernet0/0/0/11
RP/0/RP0/CPU0:router(config-if-l2)# monitor-session mon1
RP/0/RSP0/CPU0:router(config-if-mon)# mirror first 100
```

Sampled Traffic Mirroring: Example

This example shows how to configure Sampled Traffic Mirroring:

Destination Port

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:(config)# interface TenGigE 0/3/1/3
RP/0/RSP0/CPU0:(config-if)# l2transport
RP/0/RSP0/CPU0:(config-if-l2)# commit
RP/0/RSP0/CPU0:(config)# monitor-session sampled-span1
RP/0/RSP0/CPU0:(config-mon)# destination interface TenGigE 0/3/1/3
RP/0/RSP0/CPU0:(config-mon)# commit
```

Source Port

```
RP/0/RSP0/CPU0:(config)# interface TenGigE 0/3/0/0
RP/0/RSP0/CPU0:(config-if)# l2transport
RP/0/RSP0/CPU0:(config-if-l2)# monitor-session sampled-span1 direction rx-only port-level
RP/0/RSP0/CPU0:(config-if-mon)# mirror interval 512
```

```
RP/0/RSP0/CPU0:(config-if-mon)# commit
```

In order to display the session status with the Sampled SPAN information, use the **show monitor-session status detail** command.

```
RP/0/RSP0/CPU0 # show monitor-session status detail

Monitor-session sampled-span1
 Destination interface TenGigE0/3/1/3
 Source Interfaces
 -----------------
 TenGigE0/3/0/0
  Direction: Rx-only
  Port level: True
  ACL match: Disabled
  Portion:  Full packet
  Interval:  512
  Status:   Operational
```

In order to display the session statistics, use the **show monitor-session counters** command.

```
RP/0/RSP0/CPU0:router# show monitor-session counters

Monitor-session sampled-span1
 TenGigE0/3/0/0
  Rx replicated: 1952 packets, 390400 octets
  Tx replicated: 0 packets, 0 octets
  Non-replicated: 0 packets, 0 octets
```

Troubleshooting Traffic Mirroring

When you encounter any issue with traffic mirroring, begin troubleshooting by checking the output of the **show monitor-session status** command. This command displays the recorded state of all sessions and source interfaces:

```
Monitor-session sess1
<Session status>
================================================================================
Source Interface   Dir  Status
-------------------- ---- -------------------------------------------------
Gi0/0/0/0       Both <Source interface status>
Gi0/0/0/2       Both <Source interface status>
```

In the preceding example, the line marked as `<Session status>` can indicate one of these configuration errors:

Session Status	Explanation
Session is not configured globally	The session does not exist in global configuration. Check **show run** command output to ensure that a session with a correct name has been configured.

Session Status	Explanation
Destination interface <intf> is not configured	The interface that has been configured as the destination does not exist. For example, the destination interface may be configured to be a VLAN subinterface, but the VLAN subinterface may not have been yet created.
Destination interface <intf> (<down-state>)	The destination interface is not in Up state in the Interface Manager. You can verify the state using the **show interfaces** command. Check the configuration to see what might be keeping the interface from coming up (for example, a sub-interface needs to have an appropriate encapsulation configured).
Destination pseudowire is not configured	The L2VPN configuration that is to set up the pseudowire is missing. Configure the traffic mirroring session name as one segment of the xconnect p2p.
Destination pseudowire <name> (down)	The pseudowire is configured, but is down. Check the L2VPN configuration to identify why the pseudowire is not coming up.

The <Source interface status> can report these messages:

Source Interface Status	Explanation
Operational	Everything appears to be working correctly in traffic mirroring PI. Please follow up with the platform teams in the first instance, if mirroring is not operating as expected.
Not operational (Session is not configured globally)	The session does not exist in global configuration. Check the **show run** command output to ensure that a session with the right name has been configured.
Not operational (destination interface not known)	The session exists, but it either does not have a destination interface specified, or the destination interface named for the session does not exist (for example, if the destination is a sub-interface that has not been created).
Not operational (source same as destination)	The session exists, but the destination and source are the same interface, so traffic mirroring does not work.
Not operational (destination not active)	The destination interface or pseudowire is not in the Up state. See the corresponding *Session status* error messages for suggested resolution.

Source Interface Status	Explanation
Not operational (source state <down-state>)	The source interface is not in the Up state. You can verify the state using the **show interfaces** command. Check the configuration to see what might be keeping the interface from coming up (for example, a sub-interface needs to have an appropriate encapsulation configured).
Error: see detailed output for explanation	Traffic mirroring has encountered an error. Run the **show monitor-session status detail** command to display more information.

The **show monitor-session status detail** command displays full details of the configuration parameters, and of any errors encountered. For example:

RP/0/RSP0**show monitor-session status detail**

```
Monitor-session sess1
 Destination interface is not configured
 Source Interfaces
 -----------------
 GigabitEthernet0/0/0/0
  Direction: Both
  ACL match: Enabled
  Portion:  Full packet
  Status:  Not operational (destination interface not known)
 GigabitEthernet0/0/0/2
  Direction: Both
  ACL match: Disabled
  Portion:  First 100 bytes
  Status: Not operational (destination interface not known). Error: 'Viking SPAN PD' detected
 the 'warning' condition 'PRM connection creation failure'.
Monitor-session foo
 Destination next-hop GigabitEthernet 0/0/0/0
 Source Interfaces
 -----------------
 GigabitEthernet 0/1/0/0.100:
  Direction: Both
  Status:  Operating
 GigabitEthernet 0/2/0/0.200:
  Direction: Tx
  Status:  Error: <blah>

Monitor session bar
 No destination configured
 Source Interfaces
 -----------------
 GigabitEthernet 0/3/0/0.100:
  Direction: Rx
  Status:  Not operational(no destination)
```

Here are additional trace and debug commands:

```
RP/0/RSP0/CPU0:router# show monitor-session platform trace ?

 all   Turn on all the trace
 errors Display errors
 events Display interesting events
```

```
RP/0/RSP0/CPU0:router# show monitor-session trace ?

 process Filter debug by process

RP/0/RSP0/CPU0:router# debug monitor-session platform ?

 all   Turn on all the debugs
 errors VKG SPAN EA errors
 event  VKG SPAN EA event
 info  VKG SPAN EA info

RP/0/RSP0/CPU0:router# debug monitor-session platform all

RP/0/RSP0/CPU0:router# debug monitor-session platform event

RP/0/RSP0/CPU0:router# debug monitor-session platform info

RP/0/RSP0/CPU0:router# show monitor-session status ?

 detail  Display detailed output
 errors  Display only attachments which have errors
 internal Display internal monitor-session information
 |     Output Modifiers

RP/0/RSP0/CPU0:router# show monitor-session status

RP/0/RSP0/CPU0:router# show monitor-session status errors
```

Where to Go Next

When you have configured an Ethernet interface, you can configure individual VLAN subinterfaces on that Ethernet interface.

For information about modifying Ethernet management interfaces for the shelf controller (SC), route processor (RP), and distributed RP, see the Advanced Configuration and Modification of the Management Ethernet Interface on the Cisco ASR 9000 Series Router module later in this document.

For information about IPv6 see the Implementing Access Lists and Prefix Lists on

Cisco IOS XR Software module in the Cisco IOS XR IP Addresses and Services Configuration Guide.

SPAN to File

Starting with Cisco IOS XR Software Release 7.1.2, the SPAN to File is an extension of the pre-existing SPAN feature to allow network packets to be mirrored to a file instead of an interface, so that they can be analyzed later. The file format is pcap, so that it can be easily used with tools such as tcpdump or wireshark.

Note SPAN to File supports 100 local SPAN sessions, each supporting 1 destination file and 100 source interfaces. A maximum of 1000 source ports are supported across the system. Individual platforms may support lower numbers. The SPAN session may be any of these currently supported classes: Ethernet, IPv4, IPv6, MPLS-IPv4, and MPLS-IPv6.

When a file is configured as a destination for a SPAN session, a buffer is created on each node to which the network packets are logged. The buffer is for all packets on the node regardless of which interface they are from, that is, multiple interfaces may be providing packets for the same buffer. The buffers are deleted when the session configuration is removed. The file is written by each node to a location on the active RP which contains the node ID of the node on which the buffer was located.

If multiple interfaces are attached to a session, then interfaces on the same node are expected to have their packets sent to the same file. Bundle interfaces can be attached to a session with a file destination, which is similar to attaching individual interfaces. For example, this would result in a file on each node on which the bundle has members which contains packets from all members on that node.

Note The maximum traffic rate that can be mirrored for any SPAN (ingress or egress) is limited to 10% of the line rate. Exceeding this threshold may result in the loss of transit traffic.

Guidelines and Restrictions for SPAN to File

- SPAN to File feature is designed for low traffic rate (Maximum: 500 Mbps). SPAN to File may not capture all the frames when SPAN is applied on higher traffic rate.
- If the SPAN traffic exceeds 10% of the line rate, it may lead to a loss of transit traffic.
- To limit the SPAN traffic, use SPAN along with ACL.
- Immediately after the SPAN to File configuration is applied, the network processor begins the SPAN processing. This processing continues regardless of the status of the action commands explained in the following section.

Action Commands for SPAN to File

Action commands are added to start and stop network packet collection. The commands may only be run on sessions where the destination is a file. The action command auto-completes names of globally configured SPAN to File sessions. See the table below for more information on action commands.

Table 2: Action Commands for SPAN to File

Action	Command	Description
Start	`monitor-session <name> packet-collection start`	Issue this command to start writing packets for the specified session to the configured buffer.

Action	Command	Description
Stop	`monitor-session <name> packet-collection stop [ discard-data | write directory <dir> filename <filename> ]`	Issue this command to stop writing packets to the configured buffer. If the `discard-data` option is specified, the buffer is simply cleared, whereas if the `write` option is specified, the buffer is written to disk before clearing. If the buffer is to be written, it is done so in .pcap format to this location: `/<directory>/<node_id>/<filename>.pcap`. If the user adds a .pcap extension when specifying the filename, this is removed so that the extension is not added twice.

Configuring SPAN to File

Use the following command to configure SPAN to File:

```
monitor-session <name> [ethernet|ipv4|ipv6|mpls-ipv4|mpls-ipv6]
    destination file [size <kbytes>] [buffer-type linear]
```

The `monitor-session <name> [ethernet|ipv4|ipv6|mpls-ipv4|mpls-ipv6]` part of the command creates a monitor-session with the specified name and class and is a pre-existing chain point from the current SPAN feature. The `destination file [size <kbytes>] [buffer-type linear]` part of the command adds a new "file" option to the existing "destination".

`destination file` has the following configuration options:

- Buffer size.
- Two types of buffer:
  - Circular: Once the buffer is full, the start is overwritten.
  - Linear: Once the buffer is full, no further packets are logged.

Note The default buffer-type is circular. Only linear buffer is explicitly configurable. Changing any of the parameters (buffer size or type) recreates the session, and clears any buffers of packets.

All configuration options which are applied to an attachment currently supported for other SPAN types should also be supported by SPAN to file. This may include:

- ACLs
- Write only first X bytes of packet.
- Mirror interval from 512 to 16k.

Note These options are implemented by the platform when punting the packet.

Once a session has been created, then interfaces may be attached to it using the following configuration:

```
interface GigabitEthernet 0/0/0/0
    monitor-session <name> [ethernet|ipv4|ipv6|mpls-ipv4|mpls-ipv6]
```

The attachment configuration is unchanged by SPAN to File feature.

Configuration Examples

To configure a `mon1` monitor session, use the following commands:

```
monitor-session mon1 ethernet
          destination file size 230000
         !
```

In the above example, omitting the `buffer-type` option results in default circular buffer.

To configure a `mon2` monitor session, use the following commands:

```
monitor-session mon2 ethernet
          destination file size 1000 buffer-type linear
         !
```

To attach monitor session to a physical or bundle interface, use the following commands:

```
RP/0/RSP0/CPU0:router#show run interface Bundle-Ether 1
Fri Apr 24 12:12:59.348 EDT
interface Bundle-Ether1
monitor-session ms7 ethernet
!
```

Running Configuration

```
!! IOS XR Configuration 7.1.1.124I
!! Last configuration change at Tue Nov 26 19:29:05 2019 by root
!
hostname OC
logging console informational
!
monitor-session mon1 ethernet
 destination file size 230000 buffer-type circular
!
monitor-session mon2 ethernet
 destination file size 1000 buffer-type linear

!
interface Bundle-Ether1
monitor-session ms7 ethernet
end
```

Verification

To verify monitor session counters:

```
RP/0/RP0/CPU0:router#show monitor-session counters
Monitor-session mon2
  Bundle-Ether2.1
    Rx replicated: 9463555 packets, 1287043296 octets
    Tx replicated: 0 packets, 0 octets
    Non-replicated: 0 packets, 0 octets

 Monitor-session mon4
```

```
  Bundle-Ether1.1
    Rx replicated: 0 packets, 0 octets
    Tx replicated: 9732869 packets, 1284738317 octets
    Non-replicated: 0 packets, 0 octets
```

To verify packet collection status:

```
RP/0/RP0/CPU0:router#show monitor-session status
Monitor-session mon1
Destination File - Packet collecting
================================================
Source Interface      Dir   Status
--------------------- ----  ------------------
Hu0/9/0/2             Rx    Operational

Monitor-session mon2
Destination File - Packet collecting
========================================
Source Interface      Dir   Status
--------------------- ----  --------------
BE2.1                 Rx    Operational
```

If packet collection is not active, the following line is displayed:

```
Monitor-session mon2
Destination File - Not collecting
```

Introduction to File Mirroring

Prior to Cisco IOS XR Software Release 7.1.2, the router did not support file mirroring from active RP to standby RP. Administrators had to manually perform the task or use EEM scripts to sync files across active RP and standby RP. Starting with Cisco IOS XR Software Release 7.1.2, file mirroring feature enables the router to copy files or directories automatically from `/harddisk:/mirror` location in active RP to `/harddisk:/mirror` location in standby RP or RSP without user intervention or EEM scripts.

Two new CLIs have been introduced for the file mirroring feature:

- **`mirror enable`**

  The `/harddisk:/mirror` directory is created by default, but file mirroring functionality is only enabled by executing the `mirror enable` command from configuration terminal. Status of the mirrored files can be viewed with `show mirror status` command.

- **`mirror enable checksum`**

  The `mirror enable checksum` command enables MD5 checksum across active to standby RP to check integrity of the files. This command is optional.

Limitations

The following limitations apply to file mirroring:

- Supported only on Dual RP systems.
- Supports syncing only from active to standby RP. If files are copied into standby `/harddisk:/mirror` location, it won’t be synced to active RP.

- A slight delay is observed in `show mirror` command output when mirror checksum configuration is enabled.
- Not supported on multichassis systems.

Configure File Mirroring

File mirroring has to be enabled explicitly on the router. It is not enabled by default.

```
RP/0/RSP0/CPU0:router#show run mirror

Thu Jun 25 10:12:17.303 UTC
mirror enable
mirror checksum
```

Following is an example of copying running configuration to `harddisk:/mirror` location:

```
RP/0/RSP0/CPU0:router#copy running-config harddisk:/mirror/run_config
Wed Jul  8 10:25:51.064 PDT
Destination file name (control-c to abort): [/mirror/run_config]?
Building configuration..
32691 lines built in 2 seconds (16345)lines/sec
[OK]
```

Verification

To verify the syncing of file copied to mirror directory, use the `show mirror` command.

```
RP/0/RSP0/CPU0:router#show mirror
Wed Jul  8 10:31:21.644 PDT
% Mirror rsync is using checksum, this show command may take several minutes if you have
many files. Use Ctrl+C to abort
MIRROR DIR: /harddisk:/mirror/
% Last sync of this dir ended at Wed Jul  8 10:31:11 2020
Location   |Mirrored |MD5 Checksum                      |Modification Time
-------------------------------------------------------------------------
run_config |yes      |76fc1b906bec4fe08ecda0c93f6c7815 |Wed Jul  8 10:25:56 2020
```

If checksum is disabled, `show mirror` command displays the following output:

```
RP/0/RSP0/CPU0:router#show mirror
Wed Jul 8 10:39:09.646 PDT
MIRROR DIR: /harddisk:/mirror/
% Last sync of this dir ended at Wed Jul  8 10:31:11 2020
Location   |Mirrored |Modification Time
----------------------------------------
run_config |yes         |Wed Jul  8 10:25:56 2020
```

If there is a mismatch during the syncing process, use `show mirror mismatch` command to verify.

```
RP/0/RP0/CPU0:router# show mirror mismatch
Wed Jul  8 10:31:21.644 PDT
 MIRROR DIR: /harddisk:/mirror/
 % Last sync of this dir ended at Wed Jul  8 10:31:11 2020
 Location  |Mismatch Reason     |Action Needed
-----------------------------------------------
 test.txt  |newly created item. |send to standby
```