Configuring ERSPAN

This chapter describes how to configure an encapsulated remote switched port analyzer (ERSPAN) to transport mirrored traffic in an IP network on Cisco NX-OS devices.

About ERSPAN

ERSPAN transports mirrored traffic over an IP network, which provides remote monitoring of multiple switches across your network. The traffic is encapsulated at the source router and is transferred across the network. The packet is decapsulated at the destination router and then sent to the destination interface. Another method is that the destination can be the analyzer itself, which needs to understand the ERSPAN encapsulation format to parse the packet and access the inner (SPAN copy) frame.

ERSPAN Sources

The interfaces from which traffic can be monitored are called ERSPAN sources. Sources designate the traffic to monitor and whether to copy ingress, egress, or both directions of traffic. ERSPAN sources include the following:

- Ethernet ports (but not subinterfaces)
- Port channels
- The inband interface to the control plane CPU

Note When you specify the supervisor inband interface as a SPAN source, the device monitors all packets that are sent by the Supervisor CPU.

Note If you use the supervisor inband interface as a SPAN source, all packets generated by the supervisor hardware (egress) are monitored.

Rx is from the perspective of the ASIC (traffic egresses from the supervisor over the inband and is received by the ASIC/SPAN).

- VLANs
  - When a VLAN is specified as an ERSPAN source, all supported interfaces in the VLAN are ERSPAN sources.
  - VLANs can be ERSPAN sources only in the ingress direction, except for Cisco Nexus 9300-EX/-FX/-FX2 series platform switches, and Cisco Nexus 9500 series platform switches with -EX/-FX line cards.

Note A single ERSPAN session can include mixed sources in any combination of the above.

ERSPAN Sessions

You can create ERSPAN sessions that designate sources to monitor.

Localized ERSPAN Sessions

An ERSPAN session is localized when all of the source interfaces are on the same line card.

Note An ERSPAN session with a VLAN source is not localized

ERSPAN Truncation

Beginning with Cisco NX-OS Release 7.0(3)I7(1), you can configure the truncation of source packets for each ERSPAN session based on the size of the MTU. Truncation helps to decrease ERSPAN bandwidth by reducing the size of monitored packets. Any ERSPAN packet that is larger than the configured MTU size is truncated to the given size. For ERSPAN, an additional ERSPAN header is added to the truncated packet from 54 to 166 bytes depending on the ERSPAN header type. For example, if you configure the MTU as 300 bytes, the packets are replicated with an ERSPAN header size from 354 to 466 bytes depending on the ERSPAN header type configuration.

ERSPAN truncation is disabled by default. To use truncation, you must enable it for each ERSPAN session.

Prerequisites for ERSPAN

ERSPAN has the following prerequisites:

- You must first configure the ports on each device to support the desired ERSPAN configuration. For more information, see the Cisco Nexus 9000 Series NX-OS Interfaces Configuration Guide.

Guidelines and Limitations for ERSPAN

Note For scale information, see the release-specific *Cisco Nexus 9000 Series NX-OS Verified Scalability Guide*.

ERSPAN has the following configuration guidelines and limitations:

- A maximum of 48 source interfaces are supported per ERSPAN session (Rx and Tx, Rx, or Tx).
- ERSPAN destination handles jumbo frames for MTU differently based on the platform. For the following Cisco Nexus 9300 platform switches and Cisco Nexus 9500 platform switches with supporting line cards, ERSPAN destination drops the jumbo frames:
  - Cisco Nexus 9332PQ
  - Cisco Nexus 9372PX
  - Cisco Nexus 9372PX-E
  - Cisco Nexus 9372TX
  - Cisco Nexus 9372TX-E
  - Cisco Nexus 93120TX
  - Cisco Nexus 9500 platform switches with the following line cards:
    - Cisco Nexus 9564PX
    - Cisco Nexus 9464TX
    - Cisco Nexus 9464TX2
    - Cisco Nexus 9564TX
    - Cisco Nexus 9464PX
    - Cisco Nexus 9536PQ
    - Cisco Nexus 9636PQ
    - Cisco Nexus 9432PQ

  For the following Cisco Nexus 9200 platform switches and Cisco Nexus 9500 platform switches with supporting line cards, ERSPAN truncates the packets at port MTU, and issues a TX Output error:
  - Cisco Nexus 92160YC-X
  - Cisco Nexus 92304QC
  - Cisco Nexus 9272Q
  - Cisco Nexus 9232C

  - Cisco Nexus 9236C
  - Cisco Nexus 92300YC
  - Cisco Nexus 93108TC-EX
  - Cisco Nexus 93180LC-EX
  - Cisco Nexus 93180YC-EX
  - Cisco Nexus 9500 platform switches with the following line cards:
    - Cisco Nexus 9736C-EX
    - Cisco Nexus 97160YC-EX
    - Cisco Nexus 9732C-EX
    - Cisco Nexus 9732C-EXM

- For ERSPAN session limits, see the *Cisco Nexus 9000 Series NX-OS Verified Scalability Guide*.
- The number of ERSPAN sessions per line card reduces to two if the same interface is configured as a bidirectional source in more than one session.
- Only ERSPAN source sessions are supported. Destination sessions are not supported.
- Configuring two SPAN or ERSPAN sessions on the same source interface with only one filter is not supported. If the same source is used in multiple SPAN or ERSPAN sessions either all the sessions must have different filters or no sessions should have filters.
- Packets with FCS errors are not mirrored in an ERSPAN session.
- TCAM carving is not required for SPAN/ERSPAN on the following line cards:
  - Cisco Nexus 9636C-R
  - Cisco Nexus 9636Q-R
  - Cisco Nexus 9636C-RX
  - Cisco Nexus 96136YC-R

Note All other switches supporting SPAN/ERSPAN must use TCAM carving.

- Statistics are not supported for the filter access group.
- An access-group filter in an ERSPAN session must be configured as vlan-accessmap.
- Control plane packets that are generated by the supervisor cannot be ERSPAN encapsulated or filtered by an ERSPAN access control list (ACL).
- ERSPAN is not supported for management ports.
- ERSPAN does not support destinations on Layer 3 port-channel subinterfaces.
- A VLAN can be part of only one session when it is used as an ERSPAN source or filter.

- VLAN ERSPAN monitors only the traffic that leaves or enters Layer 2 ports in the VLAN.
- If you enable ERSPAN on a vPC and ERSPAN packets must be routed to the destination through the vPC, packets that come through the vPC peer link cannot be captured.
- ERSPAN is not supported over a VXLAN overlay.
- ERSPAN copies for multicast packets are made before rewrite. Therefore, the TTL, VLAN ID, any remarking due to egress policy, and so on, are not captured in the ERSPAN copy.
- The timestamp granularity of ERSPAN Type III sessions is not configurable through the CLI. It is 100 picoseconds and driven through PTP.
- ERSPAN works on default and nondefault VRFs, but ERSPAN marker packets work only on the default VRF.
- The same source can be part of multiple sessions.
- Cisco Nexus 9300-EX/FX switches cannot serve as an ERSPAN destination for Cisco Nexus 3000 and non-EX/FX Cisco Nexus 9000 switches.

The following guidelines and limitations apply to egress (Tx) ERSPAN:

- The flows for post-routed unknown unicast flooded packets are in the ERSPAN session, even if the ERSPAN session is configured to not monitor the ports on which this flow is forwarded. This limitation applies to Network Forwarding Engine (NFE) and NFE2-enabled EOR switches and ERSPAN sessions that have TX port sources.
- The following guidelines and limitations apply to ingress (Rx) ERSPAN:
  - VLAN sources are spanned only in the Rx direction.
  - Session filtering functionality (VLAN or ACL filters) is supported only for Rx sources.
  - VLANs are supported as ERSPAN sources only in the ingress direction.
- The following guidelines and limitations apply to FEX ports:
  - If the sources used in bidirectional ERSPAN sessions are from the same FEX, the hardware resources are limited to two ERSPAN sessions.
  - FEX ports are supported as ERSPAN sources in the ingress direction for all traffic and in the egress direction only for known Layer 2 unicast traffic.
  - Cisco Nexus 9300 platform switches do not support ERSPAN destination being connected on a FEX interface. The ERSPAN destination must be connected to a front panel port.
  - VLAN and ACL filters are not supported for FEX ports.
- Priority flow control (PFC) ERSPAN has the following guidelines and limitations:
  - It cannot co-exist with filters
  - It is supported only in the Rx direction on physical or port-channel interfaces. It is not supported in the Rx direction on VLAN interfaces or in the Tx direction.

Default Settings

The following table lists the default settings for ERSPAN parameters.

Table 1: Default ERSPAN Parameters

Parameters	Default
ERSPAN sessions	Created in the shut state
ERSPAN marker packet interval	100 milliseconds
Timestamp granularity of ERSPAN Type III sessions	100 picoseconds

Configuring ERSPAN

Note Be aware that the Cisco NX-OS commands for this feature may differ from those commands used in Cisco IOS.

Configuring an ERSPAN Source Session

You can configure an ERSPAN session on the local device only. By default, ERSPAN sessions are created in the shut state.

Note ERSPAN does not monitor any packets that are generated by the supervisor, regardless of their source.

Procedure

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**monitor erspan origin ip-address** *ip-address* **global** or **Example:** `switch(config)# monitor erspan origin ip-address 10.0.0.1 global`	Configures the ERSPAN global origin IP address.

	Command or Action	Purpose
Step 3	**no monitor session** {*session-number* | **all**} **Example:** `switch(config)# no monitor session 3`	Clears the configuration of the specified ERSPAN session. The new session configuration is added to the existing session configuration.

Step 4	**monitor session** {*session-number* | **all**} **type erspan-source** [**shut**]
	Example:
	`switch(config)# monitor session 3 type erspan-source`
	`switch(config-erspan-src)#`
		Configures an ERSPAN Type II source session. By default the session is bidirectional. The optional keyword shut specifies a shut state for the selected session.

Step 5	**description** *description* **Example:** `switch(config-erspan-src)# description erspan_src_session_3`	Configures a description for the session. By default, no description is defined. The description can be up to 32 alphanumeric characters.

Step 6	**source** {**interface** *type* [**tx** | **rx** |**both**] **vlan** {**number** | **range**} [**rx**]}
	Example:
	`switch(config-erspan-src)# source interface ethernet 2/1-3, ethernet 3/1 rx`
	Example:
	`switch(config-erspan-src)# source interface port-channel 2`
	Example:
	`switch(config-erspan-src)# source interface sup-eth 0 rx`
	Example:
	`switch(config-erspan-src)# source vlan 3, 6-8 rx`
	Example:
	`switch(config-erspan-src)# source interface ethernet 101/1/1-3`
		Configures the sources and traffic direction in which to copy packets. You can enter a range of Ethernet ports, a port channel, an inband interface, a range of VLANs, or a satellite port or host interface port channel on the Cisco Nexus 2000 Series Fabric Extender (FEX).
		You can configure one or more sources, as either a series of comma-separated entries or a range of numbers. You can specify the traffic direction to copy as ingress, egress, or both.
		For a unidirectional session, the direction of the source must match the direction specified in the session.
		Note Source VLANs are supported only in the ingress direction. Source FEX ports are supported in the ingress direction for all traffic and in the egress direction only for known Layer 2 unicast traffic.
		Supervisor as a source is only supported in the Rx direction.

Step 7	(Optional) Repeat Step 7 to configure all ERSPAN sources.	—
Step 8	**filter vlan** {*number* | *range*} **Example:** `switch(config-erspan-src)# filter vlan 3-5, 7`	Configures which VLANs to select from the configured sources. You can configure one or more VLANs, as either a series of comma-separated entries or a range of numbers. For information on the VLAN range,

	Command or Action	Purpose
		see the Cisco Nexus 9000 Series NX-OS Layer 2 Switching Configuration Guide. **Note** A FEX port that is configured as an ERSPAN source does not support VLAN filters.
Step 9	(Optional) Repeat Step 9 to configure all source VLANs — to filter.	—
Step 10	(Optional) **filter access-group** *acl-filter* **Example:** `switch(config-erspan-src)# filter access-group ACL1`	Associates an ACL with the ERSPAN session. (You can create an ACL using the standard ACL configuration process. For more information, see the *Cisco Nexus 9000 Series NX-OS Security Configuration Guide.*) **Note** Before executing this command, configure ip access list and associated vlan access mac. See Configuring an ERSPAN ACL.
Step 11	**destination ip** *ip-address* **Example:** `switch(config-erspan-src)# destination ip 10.1.1.1`	Configures the destination IP address in the ERSPAN session. **Note** Only one destination IP address is supported per ERSPAN source session.
Step 12	**erspan-id** *erspan-id* **Example:** `switch(config-erspan-src)# erspan-id 5`	Configures the ERSPAN ID for the ERSPAN source session. The ERSPAN range is from 1 to 1023.
Step 13	**vrf** *vrf-name* **Example:** `switch(config-erspan-src)# vrf default`	Configures the virtual routing and forwarding (VRF) instance that the ERSPAN source session uses for traffic forwarding. The VRF name can be any case-sensitive, alphanumeric string up to 32 characters.
Step 14	(Optional) **ip ttl** *ttl-number* **Example:** `switch(config-erspan-src)# ip ttl 25`	Configures the IP time-to-live (TTL) value for the ERSPAN traffic. The range is from 1 to 255.
Step 15	(Optional) **ip dscp** *dscp-number* **Example:** `switch(config-erspan-src)# ip dscp 42`	Configures the differentiated services code point (DSCP) value of the packets in the ERSPAN traffic. The range is from 0 to 63.
Step 16	(Optional) [**no**] **marker-packet** *milliseconds* **Example:**	Enables the ERSPAN marker packet for a session in order to recover the real value of the ERSPAN timestamp. The interval can range

	Command or Action	Purpose
	`switch(config-erspan-src)# marker-packet 100`	from 100 to 1000 milliseconds. The **no** form of this command disables the marker packet for the session. **Note** ERSPAN marker packets only apply to Type 3 sessions.
Step 17	**no shut** **Example:** `switch(config-erspan-src)# no shut`	Enables the ERSPAN source session. By default, the session is created in the shut state.

Step 18	**exit**
	Example:
	`switch(config-erspan-src)# exit`
	`switch(config)#`
		Exits the monitor configuration mode.

Step 19	(Optional) **show monitor session** {**all** | *session-number* | **range** *session-range*} [**brief**] **Example:** `switch(config)# show monitor session 3`	Displays the ERSPAN session configuration.
Step 20	(Optional) **show running-config monitor** **Example:** `switch(config)# show running-config monitor`	Displays the running ERSPAN configuration.
Step 21	(Optional) **show startup-config monitor** **Example:** `switch(config)# show startup-config monitor`	Displays the ERSPAN startup configuration.
Step 22	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Shutting Down or Activating an ERSPAN Session

You can shut down ERSPAN sessions to discontinue the copying of packets from sources to destinations. You can shut down one session in order to free hardware resources to enable another session. By default, ERSPAN sessions are created in the shut state.

You can enable ERSPAN sessions to activate the copying of packets from sources to destinations. To enable an ERSPAN session that is already enabled but operationally down, you must first shut it down and then enable it. You can shut down and enable the ERSPAN session states with either a global or monitor configuration mode command.

Procedure

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**monitor session** {*session-range* | **all**} **shut** **Example:** `switch(config)# monitor session 3 shut`	Shuts down the specified ERSPAN sessions. By default, sessions are created in the shut state.
Step 3	**no monitor session** {*session-range* | **all**} **shut** **Example:** `switch(config)# no monitor session 3 shut`	Resumes (enables) the specified ERSPAN sessions. By default, sessions are created in the shut state. If a monitor session is enabled but its operational status is down, then to enable the session, you must first specify the **monitor session shut** command followed by the **no monitor session shut** command.

Step 4	**monitor session** *session-number* **type erspan-source**
	Example:
	`switch(config)# monitor session 3 type erspan-source`
	`switch(config-erspan-src)#`
		Enters the monitor configuration mode for the ERSPAN source type. The new session configuration is added to the existing session configuration.

Step 5	**shut** **Example:** `switch(config-erspan-src)# shut`	Shuts down the ERSPAN session. By default, the session is created in the shut state.
Step 6	**no shut** **Example:** `switch(config-erspan-src)# no shut`	Enables the ERSPAN session. By default, the session is created in the shut state.

Step 7	**exit**
	Example:
	`switch(config-erspan-src)# exit`
	`switch(config)#`
		Exits the monitor configuration mode.

Step 8	(Optional) **show monitor session all** **Example:** `switch(config)# show monitor session all`	Displays the status of ERSPAN sessions.
Step 9	(Optional) **show running-config monitor** **Example:**	Displays the ERSPAN running configuration.

	Command or Action	Purpose
	`switch(config)# show running-config monitor`	
Step 10	(Optional) **show startup-config monitor** **Example:** `switch(config)# show startup-config monitor`	Displays the ERSPAN startup configuration.
Step 11	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Configuring an ERSPAN ACL

You can create an IPv4 ERSPAN ACL on the device and add rules to it.

Before you begin

To modify the DSCP value or the GRE protocol, you need to allocate a new destination monitor session. A maximum of four destination monitor sessions are supported.

Procedure

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**ip access-list** *acl-name*
	Example:
	`switch(config)# ip access-list erspan-acl`
	`switch(config-acl)#`
		Creates the ERSPAN ACL and enters IP ACL configuration mode. The *acl-name* argument can be up to 64 characters.

Step 3	[*sequence-number*] {**permit** | **deny**} *protocol source destination* [**set-erspan-dscp** *dscp-value*] [**set-erspan-gre-proto** *protocol-value*] **Example:** `switch(config-acl)# permit ip 192.168.2.0/24 any set-erspan-dscp 40 set-erspan-gre-proto 5555` **Example:**	Creates a rule in the ERSPAN ACL. You can create many rules. The *sequence-number* argument can be a whole number between 1 and 4294967295. The **permit** and **deny** commands support many ways of identifying traffic. The **set-erspan-dscp** option sets the DSCP value in the ERSPAN outer IP header. The range for the DSCP value is from 0 to 63. The DSCPvalue configured in the ERSPAN ACL

	Command or Action	Purpose

	`switch(config)# ip access-list match_11_pkts`
	`switch(config-acl)# permit ip 10.0.0.0/24 any`
	`switch(config-acl)# exit`
		overrides the value configured in the monitorsession. If you do not include this option in the ERSPAN ACL, 0 or the DSCP value configured in the monitor session will be set.
		The **set-erspan-gre-proto** option sets the protocol value in the ERSPAN GRE header. The range for the protocol value is from 0 to 65535. If you do not include this option in the ERSPAN ACL, the default value of 0x88be will be set as the protocol in the GRE header for ERSPAN-encapsulated packets
		Each access control entry (ACE) with the s**et-erspan-gre-proto** or **set-erspan-dscp** action consumes one destination monitorsession. A maximum of three ACEs with one of these actions is supported per ERSPAN ACL. For example, you can configure one of the following:
		• One ERSPAN session with an ACL having a maximum of three ACEs with the **set-erspan-gre-proto** or **set-erspan-dscp** action
		• One ERSPAN session with an ACL having two ACEs with the **set-erspan-gre-proto** or **set-erspan-dscp** action and one additional local or ERSPAN session
		• A maximum of two ERSPAN sessions with an ACL having one ACE with the **set-erspan-gre-proto** or **set-erspan-dscp** action

Step 4	**show ip access-lists** *name* **Example:** `switch(config-acl)# show ip access-lists erpsan-acl`	Displays the ERSPAN ACL configuration.
Step 5	**show monitor session** {**all** | *session-number* | **range** *session-range*} [**brief**] **Example:** `switch(config-acl)# show monitor session 1`	Displays the ERSPAN session configuration.
Step 6	(Optional) **copy running-config startup-config** **Example:** `switch(config-acl)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Configuring UDF-Based ERSPAN

You can configure the device to match on user-defined fields (UDFs) of the outer or inner packet fields (header or payload) and to send the matching packets to the ERSPAN destination. Doing so can help you to analyze and isolate packet drops in the network.

Before you begin

Make sure that the appropriate TCAM region (racl, ifacl, or vacl) has been configured using the **hardware access-list tcam region** command to provide enough free space to enable UDF-based ERSPAN. For information, see the "Configuring ACL TCAM Region Sizes" section in the Cisco Nexus 9000 Series NX-OS Security Configuration Guide.

Procedure

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**udf** *udf-name offset-base offset length*
	Example:
	`switch(config)# udf udf-x packet-start 12 1`
	`switch(config)# udf udf-y header outer l3 20 2`
		Defines the UDF as follows:
		• *udf-name*—Specifies the name of the UDF. You can enter up to 16 alphanumeric characters for the name.
		• *offset-base*—Specifies the UDF offset base as follows, where **header** is the packet header to consider for the offset: **packet-start** | **header** {**outer** | **inner** {**l3** | **l4**}}.
		• *offset*—Specifies the number of bytes offset from the offset base. To match the first byte from the offset base (Layer 3/Layer 4 header), configure the offset as 0.
		• *length*—Specifies the number of bytes from the offset. Only 1 or 2 bytes are supported. To match additional bytes, you must define multiple UDFs.
		You can define multiple UDFs, but Cisco recommends defining only required UDFs.

Step 3	**hardware access-list tcam region** {**racl** | **ifacl** | **vacl** }**qualify udf** *udf-names* **Example:**	Attaches the UDFs to one of the following TCAM regions: • racl—Applies to Layer 3 ports.—Applies to layer 2 and Layer 3 ports.

	Command or Action	Purpose
	`switch(config)# hardware access-list tcam region racl qualify udf udf-x udf-y`	• ifacl—Applies to Layer 2 ports. • vacl—Applies to source VLANs. You can attach up to 8 UDFs to a TCAM region. **Note** When the UDF qualifier is added, the TCAM region goes from single wide to double wide. Make sure enough free space is available; otherwise, this command will be rejected. If necessary, you can reduce the TCAM space from unused regions and then re-enter this command. For more information, see the "Configuring ACL TCAM Region Sizes" section in the Cisco Nexus 9000 Series NX-OS Security Configuration Guide. **Note** The **no** form of this command detaches the UDFs from the TCAM region and returns the region to single wide.
Step 4	Required: **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.
Step 5	Required: **reload** **Example:** `switch(config)# reload`	Reloads the device. **Note** Your UDF configuration is effective only after you enter **copy running-config startup-config** + **reload**.

Step 6	**ip access-list** *erspan-acl*
	Example:
	`switch(config)# ip access-list erspan-acl-udf-only`
	`switch(config-acl)#`
		Creates an IPv4 access control list (ACL) and enters IP access list configuration mode.

Step 7	Enter one of the following commands: • **permit udf** *udf-name value mask* • **permit ip** *source destination* **udf** *udf-name value mask*	Configures the ACL to match only on UDFs (example 1) or to match on UDFs along with the current access control entries (ACEs) for the outer packet fields (example 2).

	Command or Action	Purpose

	Example:
	`switch(config-acl)# permit udf udf-x 0x40 0xF0 udf-y 0x1001 0xF00F`
	Example:
	`switch(config-acl)# permit ip 10.0.0.0/24 any udf udf-x 0x02 0x0F udf-y 0x1001 0xF00F`
		A single ACL can have ACEs with and without UDFs together. Each ACE can have different UDF fields to match, or all ACEs can match for the same list of UDFs.

Step 8	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Configuring ERSPAN Truncation

You can configure truncation for local and ERSPAN source sessions only.

Procedure

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**monitor session** *session-number* **type erspan-source**
	Example:
	`switch(config)# monitor session 10 type erspan-source`
	`switch(config-erspan-src)#`
		Enters monitor configuration mode for the specified ERSPAN session.

Step 3	**source interface** *type slot*/*port* [**rx** | **tx** | **both**] **Example:** `switch(config-erspan-src)# source interface ethernet 1/5 both`	Configures the source interface.

Step 4	**mtu** *size*
	Example:
	`switch(config-erspan-src)# mtu 512`
	Example:
	`switch(config-erspan-src)# mtu ?`
	`<512-1518> Enter the value of MTU truncation size for ERSPAN packets (erspan header + truncated original packet)`
		Configures the MTU size for truncation. Any ERSPAN packet that is larger than the configured MTU size is truncated to the configured size. The MTU ranges for ERSPAN packet truncation are:
		• The MTU size range is 512 to 1518 bytes for Cisco Nexus 9300-EX platform switches.

	Command or Action	Purpose
		• The MTU size range is 64 to 1518 bytes for Cisco Nexus 9300-FX platform switches.
Step 5	**destination interface** *type slot/port* **Example:** `switch(config-erspan-src)# destination interface Ethernet 1/39`	Configures the Ethernet ERSPAN destination port.
Step 6	**no shut** **Example:** `switch(config-erspan-src)# no shut`	Enables the ERSPAN session. By default, the session is created in the shut state.
Step 7	(Optional) **show monitor session** *session* **Example:** `switch(config-erspan-src)# show monitor session 5`	Displays the ERSPAN configuration.
Step 8	**copy running-config startup-config** **Example:** `switch(config-erspan-src)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Verifying the ERSPAN Configuration

To display the ERSPAN configuration, perform one of the following tasks:

Command	Purpose
show ip access-lists *name*	Displays the ERSPAN ACL configuration.

Command	Purpose
show monitor session {**all** | *session-number* | **range** *session-range*} [**brief**]	Displays the ERSPAN session configuration. The output includes the egress interface that is used to send the ERSPAN packets. The output varies depending on the type of egress interface used: • Physical Layer 3 interface—Displays the interface name. • SVI interface—Displays the member interface through which the route was learned. • Layer 3 port channel—Displays the port-channel interface name. • Layer 3 subinterface—Displays the parent interface name. • ECMP path—Displays the name of one of the equal-cost multipath (ECMP) member interfaces. Only the interface that is displayed will be used for mirroring the traffic even though the route is ECMP. • PFC on interfaces—Displays the priority flow control (PFC) status on the interface.
show running-config monitor	Displays the running ERSPAN configuration.
show startup-config monitor	Displays the ERSPAN startup configuration.

Configuration Examples for ERSPAN

Configuration Example for a Unidirectional ERSPAN Session

This example shows how to configure a unidirectional ERSPAN session:

```
switch# configure terminal
switch(config)# interface ethernet 14/30
switch(config-if)# no shut
switch(config-if)# exit
switch(config)# no monitor session 3
switch(config)# monitor session 3 rxswitch(config-erspan-src)# source interface ethernet
2/1-3 rx
switch(config-erspan-src)# erspan-id 1
switch(config-erspan-src)# ip ttl 16
switch(config-erspan-src)# ip dscp 5
switch(config-erspan-src)# vrf default
switch(config-erspan-src)# destination ip 10.1.1.2
switch(config-erspan-src)# no shut
switch(config-erspan-src)# exit
switch(config)# show monitor session 1
```

Configuration Example for an ERSPAN ACL

This example shows how to configure an ERSPAN ACL:

```
switch# configure terminal
switch(config)# ip access-list match_10_pkts
switch(config-acl)# permit ip 10.0.0.0/24 any
switch(config-acl)# exit
switch(config)# ip access-list match_172_pkts
switch(config-acl)# permit ip 172.16.0.0/24 any
switch(config-acl)# exit
```

In the case of different ERSPAN destinations where the interesting traffic is chosen based on the defined ACL filters, the last configured session would always have the higher priority.

For example, if Monitor Session 1 is configured; then Monitor Session 2 is configured; then ERSPAN traffic filter works as intended. But, if the user goes back to Monitor Session 1 and re-applies one of the existing configuration line (no new changes in the config); then the spanned traffic switches back to Monitor Session 1.

Configuration Example for a Marker Packet

This example shows how to enable the ERSPAN marker packet with an interval of 2 seconds:

```
switch# configure terminal
switch(config)# monitor erspan origin ip-address 172.28.15.250 global
switch(config)# monitor session 1 type erspan-source
switch(config-erspan-src)# header-type 3
switch(config-erspan-src)# erspan-id 1
switch(config-erspan-src)# ip ttl 16
switch(config-erspan-src)# ip dscp 5
switch(config-erspan-src)# vrf default
switch(config-erspan-src)# destination ip 10.1.1.2
switch(config-erspan-src)# source interface ethernet 1/15 both
switch(config-erspan-src)# marker-packet 100
switch(config-erspan-src)# no shut
switch(config-erspan-src)# show monitor session 1
session 1
---------------
type              : erspan-source
state             : up
granularity       : nanoseconds
erspan-id         : 1
vrf-name          : default
destination-ip    : 10.1.1.2
ip-ttl            : 16
ip-dscp           : 5
header-type       : 3
origin-ip         : 172.28.15.250 (global)
source intf       :
    rx            : Eth1/15
    tx            : Eth1/15
    both          : Eth1/15
    rx            :
marker-packet     : enabled
packet interval   : 100
packet sent       : 25
packet failed     : 0
egress-intf       :
```

Configuration Examples for UDF-Based ERSPAN

This example shows how to configure UDF-based ERSPAN to match on the inner TCP flags of an encapsulated IP-in-IP packet using the following match criteria:

- Outer source IP address: 10.0.0.2
- Inner TCP flags: Urgent TCP flag is set
- Bytes: Eth Hdr (14) + Outer IP (20) + Inner IP (20) + Inner TCP (20, but TCP flags at 13th byte)
- Offset from packet-start: 14 + 20 + 20 + 13 = 67
- UDF match value: 0x20
- UDF mask: 0xFF

```
udf udf_tcpflags packet-start 67 1
hardware access-list tcam region racl qualify udf udf_tcpflags
copy running-config startup-config
reload
ip access-list acl-udf
permit ip 10.0.0.2/32 any udf udf_tcpflags 0x20 0xff
monitor session 1 type erspan-source
source interface Ethernet 1/1
filter access-group acl-udf
```

This example shows how to configure UDF-based ERSPAN to match regular IP packets with a packet signature (DEADBEEF) at 6 bytes after a Layer 4 header start using the following match criteria:

- Outer source IP address: 10.0.0.2
- Inner TCP flags: Urgent TCP flag is set
- Bytes: Eth Hdr (14) + IP (20) + TCP (20) + Payload: 112233445566DEADBEEF7788
- Offset from Layer 4 header start: 20 + 6 = 26
- UDF match value: 0xDEADBEEF (split into two-byte chunks and two UDFs)
- UDF mask: 0xFFFFFFFF

```
udf udf_pktsig_msb header outer l3 26 2
udf udf_pktsig_lsb header outer l3 28 2
hardware access-list tcam region racl qualify udf udf_pktsig_msb udf_pktsig_lsb
copy running-config startup-config
reload
ip access-list acl-udf-pktsig
permit udf udf_pktsig_msb 0xDEAD 0xFFFF udf udf_pktsig_lsb 0xBEEF 0xFFFF
monitor session 1 type erspan-source
source interface Ethernet 1/1
filter access-group acl-udf-pktsig
```

Configuration Example for ERSPAN Truncation

This example shows how to configure ERSPAN truncation for use with MPLS stripping:

```
mpls strip
ip access-list mpls
  statistics per-entry
```

```
  20 permit ip any any redirect Ethernet1/5

interface Ethernet1/5
  switchport
  switchport mode trunk
  mtu 9216
  no shutdown

monitor session 1
  source interface Ethernet1/5 tx
  mtu 64
  destination interface Ethernet1/6
  no shut
monitor session 21 type erspan-source
  description "ERSPAN Session 21"
  header-type 3
  erspan-id 21
  vrf default
  destination ip 10.1.1.2
  source interface Ethernet1/5 tx
  mtu 64
  no shut
monitor session 22 type erspan-source
  description "ERSPAN Session 22"
  erspan-id 22
  vrf default
  destination ip 10.2.1.2
  source interface Ethernet1/5 tx
  mtu 750
  no shut
monitor session 23 type erspan-source
  description "ERSPAN Session 23"
  header-type 3
  marker-packet 1000
  erspan-id 23
  vrf default
  destination ip 10.3.1.2
  source interface Ethernet1/5 tx
  mtu 1000
  no shut
```