



## E Commands

---

This chapter describes the Cisco NX-OS Ethernet and virtual Ethernet commands that begin with E.

# encapsulation dot1Q

To enable IEEE 802.1Q encapsulation of traffic on a specified subinterface, use the **encapsulation dot1q** command. To disable encapsulation, use the **no** form of this command.

**encapsulation dot1Q** *vlan-id*

**no encapsulation dot1Q** *vlan-id*

<b>Syntax Description</b>	<i>vlan-id</i>	VLAN to set when the interface is in access mode; valid values are from 1 to 4093, except for the VLANs reserved for internal switch use.
---------------------------	----------------	---

<b>Command Default</b>	No encapsulation
------------------------	------------------

<b>Command Modes</b>	Subinterface configuration mode
----------------------	---------------------------------

<b>Command History</b>	<b>Release</b>	<b>Modification</b>
	5.2(1)N1(1)	This command was introduced.

**Usage Guidelines** IEEE 802.1Q encapsulation is configurable on Ethernet and EtherChannel interfaces. IEEE 802.1Q is a standard protocol for interconnecting multiple switches and routers and for defining VLAN topologies. Use the **encapsulation dot1q** command in subinterface range configuration mode to apply a VLAN ID to the subinterface.



**Note**

This command is not applicable to loopback interfaces.

This command does not require a license.

**Examples** This example shows how to enable dot1Q encapsulation on a subinterface for VLAN 30:

```
switch(config)# interface ethernet 1/5.1
switch(config-subif)# encapsulation dot1q 30
switch(config-subif)#
```

<b>Related Commands</b>	<b>Command</b>	<b>Description</b>
	<b>show vlan dot1Q</b>	Displays dot1Q encapsulation information for a VLAN.

# errdisable detect cause

To enable error-disable (err-disabled) detection in an application, use the **errdisable detect cause** command. To disable error disable detection, use the **no** form of this command.

**errdisable detect cause {all | link-flap | loopback}**

**no errdisable detect cause {all | link-flap | loopback}**

Syntax Description	all	Enables error detection on all cases.
	<b>link-flap</b>	Enables error disable detection on linkstate-flapping.
	<b>loopback</b>	Enables error disable detection on loopback.

**Command Default** Enabled

**Command Modes** Global configuration mode

Command History	Release	Modification
	5.2(1)N1(1)	This command was introduced.

**Usage Guidelines** When error disable detection is enabled and a cause is detected on an interface, the interface is placed in an err-disabled state, which is an operational state that is similar to the link-down state.

**Examples** This example shows how to enable the err-disabled detection on linkstate-flapping:

```
switch(config)# errdisable detect cause link-flap
switch(config)#
```

Related Commands	Command	Description
	<b>errdisable recovery</b>	Configures recovery from the err-disabled state.
	<b>show interface status err-disabled</b>	Displays the interface error disabled state.

## errdisable recovery cause

To configure the application to bring the interface out of the error-disabled (err-disabled) state and retry coming up, use the **errdisable recovery cause** command. To revert to the defaults, use the **no** form of this command.

```
errdisable recovery cause {all | bpduguard | failed-port-state | link-flap-recovery |
  pause-rate-limit | uddl}
```

```
no errdisable recovery cause {all | bpduguard | failed-port-state | link-flap-recovery |
  pause-rate-limit | uddl}
```

### Syntax Description

<b>all</b>	Enables a timer to recover from all causes.
<b>bpduguard</b>	Enables a timer to recover from bridge protocol data unit (BPDU) Guard error disable state.
<b>failed-port-state</b>	Enables a timer to recover from a Spanning Tree Protocol (STP) set port state failure.
<b>link-flap</b>	Enables a timer to recover from linkstate flapping.
<b>pause-rate-limit</b>	Enables a timer to recover from the pause rate limit error disabled state.
<b>uddl</b>	Enables a timer to recover from the Unidirectional Link Detection (UDLD) error disabled state.

### Command Default

None

### Command Modes

Global configuration mode

### Command History

Release	Modification
5.2(1)N1(1)	This command was introduced.

### Usage Guidelines

When error disable recovery is enabled, the interface automatically recovers from the err-disabled state, and the device retries bringing the interface up.

### Examples

This example shows how to enable error disable recovery from linkstate-flapping:

```
switch(config)# errdisable recovery cause link-flap
switch(config)#
```

**Related Commands**

<b>Command</b>	<b>Description</b>
<b>errdisable detect cause</b>	Enables the error disabled (err-disabled) detection.
<b>show interface status err-disabled</b>	Displays the interface error disabled state.

# errdisable recovery interval

To configure the recovery time interval to bring the interface out of the error-disabled (err-disabled) state, use the **errdisable recovery interval** command. To revert to the defaults, use the **no** form of this command.

**errdisable recovery interval** *time*

**no errdisable recovery interval**

<b>Syntax Description</b>	<i>time</i>	Error disable recovery time interval. The range is from 30 to 65535 seconds.
---------------------------	-------------	--

<b>Command Default</b>	Disabled
------------------------	----------

<b>Command Modes</b>	Global configuration mode
----------------------	---------------------------

<b>Command History</b>	<b>Release</b>	<b>Modification</b>
	5.2(1)N1(1)	This command was introduced.

<b>Usage Guidelines</b>	<p>When error disable recovery is enabled, the interface automatically recovers from the err-disabled state, and the device retries bringing the interface up.</p> <p>The device waits 300 seconds to retry.</p>
-------------------------	--

<b>Examples</b>	This example shows how to enable error disable recovery time interval to 100 seconds:
-----------------	---

```
switch(config)# errdisable recovery interval 100
switch(config)#
```

<b>Related Commands</b>	<b>Command</b>	<b>Description</b>
	<b>errdisable recovery cause</b>	Enables an error disabled recovery on an interface.
<b>show interface status err-disabled</b>	Displays the interface error disabled state.	

# erspan-id

To configure the flow ID for an Encapsulated Remote Switched Port Analyzer (ERSPAN)) session, use the **erspan-id** command. To remove the flow ID, use the **no** form of this command.

```
erspan-id flow_id
```

<b>Syntax Description</b>	<i>flow_id</i>	ERSPAN flow ID. The range is from 1 to 1023.
---------------------------	----------------	--

<b>Command Default</b>	None
------------------------	------

<b>Command Modes</b>	ERSPAN session configuration mode
----------------------	-----------------------------------

<b>Command History</b>	<b>Release</b>	<b>Modification</b>
	5.2(1)N1(1)	This command was introduced.

<b>Usage Guidelines</b>	This command does not require a license.
-------------------------	--

**Examples** This example shows how to configure the flow ID for an ERSPAN session:

```
switch# configure terminal
switch(config)# monitor session 1 type erspan-source
switch(config-erspan-src)# erspan-id 100
switch(config-erspan-src)#
```

<b>Related Commands</b>	<b>Command</b>	<b>Description</b>
		<b>ip dscp</b>
	<b>ip ttl</b>	Configures the IP time-to-live (TTL) value of the ERSPAN traffic.
	<b>mtu</b>	Sets the maximum transmission unit (MTU) size for SPAN packet.
	<b>vrf</b>	Configures the VRF for ERSPAN traffic forwarding.
	<b>monitor-session</b>	Enters the monitor configuration mode for configuring an ERSPAN or SPAN session for analyzing traffic between ports.

# extension-key

To configure the extension key to be used to connect to the vCenter Server, use the **extension-key** command.

**extension-key** *extn-ID*

<b>Syntax Description</b>	<i>extn-ID</i>	Extension ID. The ID can be a maximum of 80 alphanumeric characters.
---------------------------	----------------	--

<b>Command Default</b>	None
------------------------	------

<b>Command Modes</b>	SVS connection configuration mode
----------------------	-----------------------------------

<b>Command History</b>	<b>Release</b>	<b>Modification</b>
	5.2(1)N1(1)	This command was introduced.

<b>Usage Guidelines</b>	This command does not require a license.
-------------------------	--

<b>Examples</b>	This example shows how to configure the extension key for a vCenter Server:
-----------------	---

```
switch# configure terminal
switch(config)# svs connection SVSConn
switch(config-svs-conn)# extension-key vckey
switch(config-svs-conn)#
```

<b>Related Commands</b>	<b>Command</b>	<b>Description</b>
	<b>show svs connections</b>	Displays SVS connection information.
<b>svs connection</b>	Enables an SVS connection.	