Interface and Hardware Components Configuration Guide, Cisco IOS XE Gibraltar 16.12.x (Catalyst 3850 Switches)

First Published: 2019-07-31

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED “AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

All printed copies and duplicate soft copies of this document are considered uncontrolled. See the current online version for the latest version.

Cisco has more than 200 offices worldwide. Addresses and phone numbers are listed on the Cisco website at www.cisco.com/go/offices.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: https://www.cisco.com/c/en/us/about/legal/trademarks.html. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1721R)

© 2019 Cisco Systems, Inc. All rights reserved.

CONTENTS

CHAPTER 1 **Configuring Interface Characteristics** 1

Information About Configuring Interface Characteristics 1

Interface Types 1

Port-Based VLANs 1

Switch Ports 2

Routed Ports 3

Switch Virtual Interfaces 3

EtherChannel Port Groups 4

10-Gigabit Ethernet Interfaces 5

Multigigabit Ethernet 5

Power over Ethernet 5

Using the Switch USB Ports 6

USB Mini-Type B Console Port 6

Interface Connections 7

Interface Configuration Mode 7

Default Ethernet Interface Configuration 8

Interface Speed and Duplex Mode 10

Speed and Duplex Configuration Guidelines 10

IEEE 802.3x Flow Control 11

Layer 3 Interfaces 11

How to Configure Interface Characteristics 12

Configuring Interfaces 12

Adding a Description for an Interface 13

Configuring a Range of Interfaces 14

Configuring and Using Interface Range Macros 16

Configuring Multigigabit Ethernet Parameters 17

Configuring IEEE 802.3x Flow Control 18
Configuring Layer 3 Interfaces 19
Configuring Logical Layer 3 GRE Tunnel Interfaces 20
Configuring SVI Autostate Exclude 22
Shutting Down and Restarting the Interface 23
Configuring the Console Media Type 24
Configuring the USB Inactivity Timeout 25
Monitoring Interface Characteristics 26
Monitoring Interface Status 26
Clearing and Resetting Interfaces and Counters 27
Configuration Examples for Interface Characteristics 27
Adding a Description to an Interface: Example 27
Displaying Downshift Status of Interfaces: Examples 27
Identifying Interfaces on a Stack-Capable Switch: Examples 28
Configuring a Range of Interfaces: Examples 28
Configuring and Using Interface Range Macros: Examples 29
Setting Interface Speed and Duplex Mode: Example 29
Configuring Layer 3 Interfaces: Example 29
Configuring the Console Media Type: Example 29
Configuring the USB Inactivity Timeout: Example 30
Additional References for the Interface Characteristics Feature 30
Feature History and Information for Configuring Interface Characteristics 31

CHAPTER 2 **Configuring Auto-MDIX** 33
Prerequisites for Auto-MDIX 33
Restrictions for Auto-MDIX 33
Information About Configuring Auto-MDIX 34
Auto-MDIX on an Interface 34
How to Configure Auto-MDIX 34
Configuring Auto-MDIX on an Interface 34
Example for Configuring Auto-MDIX 35
Additional References for Auto-MDIX 36
Feature History and Information for Auto-MDIX 36

CHAPTER 3 **Configuring Ethernet Management Port** 37

Prerequisites for Ethernet Management Ports 37

Information About the Ethernet Management Port 37

Ethernet Management Port Direct Connection to a Device 37

Ethernet Management Port Connection to Stack Devices using a Hub 38

Ethernet Management Port and Routing 38

Supported Features on the Ethernet Management Port 39

How to Configure the Ethernet Management Port 40

Disabling and Enabling the Ethernet Management Port 40

Additional References for Ethernet Management Ports 41

Feature History and Information for Ethernet Management Ports 41

CHAPTER 4 **Configuring LLDP, LLDP-MED, and Wired Location Service** 43

Restrictions for LLDP 43

Information About LLDP, LLDP-MED, and Wired Location Service 43

LLDP 43

LLDP Supported TLVs 44

LLDP and Cisco Device Stacks 44

LLDP-MED 44

LLDP-MED Supported TLVs 44

Wired Location Service 46

Default LLDP Configuration 47

How to Configure LLDP, LLDP-MED, and Wired Location Service 47

Enabling LLDP 47

Configuring LLDP Characteristics 48

Configuring LLDP-MED TLVs 50

Configuring Network-Policy TLV 51

Configuring Location TLV and Wired Location Service 54

Enabling Wired Location Service on the Device 56

Configuration Examples for LLDP, LLDP-MED, and Wired Location Service 57

Configuring Network-Policy TLV: Examples 57

Monitoring and Maintaining LLDP, LLDP-MED, and Wired Location Service 58

Additional References for LLDP, LLDP-MED, and Wired Location Service 59

Feature Information for LLDP, LLDP-MED, and Wired Location Service 59

CHAPTER 5 **Configuring System MTU 61**

Restrictions for System MTU 61

Information About the MTU 61

System MTU Value Application 61

How to Configure MTU Sizes 62

Configuring the System MTU 62

Configuring Protocol-Specific MTU 63

Configuration Examples for System MTU 64

Example: Configuring Protocol-Specific MTU 64

Example: Configuring the System MTU 64

Additional References for System MTU 64

Feature Information for System MTU 65

CHAPTER 6 **Configuring Internal Power Supplies 67**

Information About Internal Power Supplies 67

How to Configure Internal Power Supplies 67

Configuring Internal Power Supply 67

Monitoring Internal Power Supplies 68

Configuration Examples for Internal Power Supplies 68

Additional References for Internal Power Supplies 69

Feature History and Information for Internal Power Supplies 70

CHAPTER 7 **Configuring Power over Ethernet 71**

Information About PoE 71

PoE and PoE+ Ports 71

Supported Protocols and Standards 71

Powered-Device Detection and Initial Power Allocation 72

Power Management Modes 73

Cisco Universal Power Over Ethernet 76

How to Configure PoE and UPoE 76

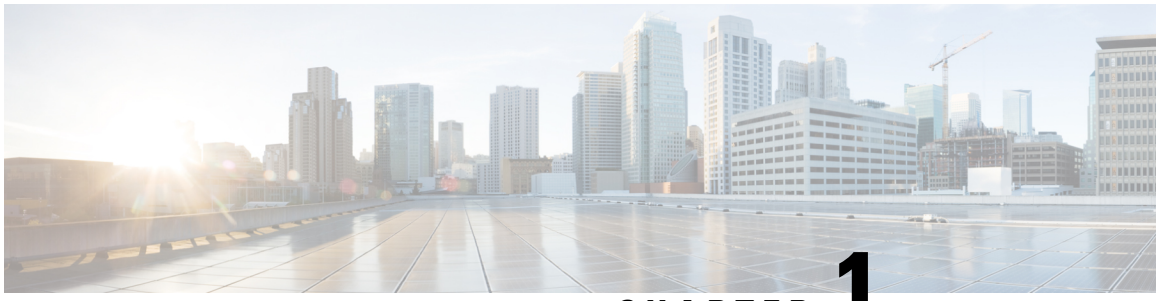
Configuring a Power Management Mode on a PoE Port 76

Enabling Power on Signal/Spare Pairs 78

Configuring Power Policing 78
Monitoring Power Status 81
Additional References 81
Feature Information for PoE 82

CHAPTER 8 Configuring the Cisco Expandable Power System 2200 83
Restrictions for Configuring the XPS 2200 83
Information About Configuring the XPS 2200 83
Cisco eXpandable Power System (XPS) 2200 Overview 83
XPS 2200 Power Supply Modes 84
RPS Mode 84
Stack Power Mode 85
Mixed Modes 86
XPS 2200 System Defaults 86
How to Configure the XPS 2200 87
Configuring System Names 87
Configuring XPS Ports 88
Configuring XPS Power Supplies 89
Monitoring and Maintaining the XPS 2200 90
Additional References for XPS 2200 90
Feature History and Information for XPS 2200 91

CHAPTER 9 Configuring EEE 93
Restrictions for EEE 93
Information About EEE 93
EEE Overview 93
Default EEE Configuration 93
How to Configure EEE 94
Enabling or Disabling EEE 94
Monitoring EEE 95
Configuration Examples for Configuring EEE 96
Additional References for EEE 96
Feature Information for Configuring EEE 96



CHAPTER 1

Configuring Interface Characteristics

- Information About Configuring Interface Characteristics, on page 1
- How to Configure Interface Characteristics, on page 12
- Monitoring Interface Characteristics, on page 26
- Configuration Examples for Interface Characteristics, on page 27
- Additional References for the Interface Characteristics Feature, on page 30
- Feature History and Information for Configuring Interface Characteristics, on page 31

Information About Configuring Interface Characteristics

Interface Types

This section describes the different types of interfaces supported by the device. The rest of the chapter describes configuration procedures for physical interface characteristics.

Note The stack ports on the rear of the stacking-capable devices are not Ethernet ports and cannot be configured.

Port-Based VLANs

A VLAN is a switched network that is logically segmented by function, team, or application, without regard to the physical location of the users. Packets received on a port are forwarded only to ports that belong to the same VLAN as the receiving port. Network devices in different VLANs cannot communicate with one another without a Layer 3 device to route traffic between the VLANs.

VLAN partitions provide hard firewalls for traffic in the VLAN, and each VLAN has its own MAC address table. A VLAN comes into existence when a local port is configured to be associated with the VLAN, when the VLAN Trunking Protocol (VTP) learns of its existence from a neighbor on a trunk, or when a user creates a VLAN. VLANs can be formed with ports across the stack.

To configure VLANs, use the **vlan** *vlan-id* global configuration command to enter VLAN configuration mode. The VLAN configurations for normal-range VLANs (VLAN IDs 1 to 1005) are saved in the VLAN database. If VTP is version 1 or 2, to configure extended-range VLANs (VLAN IDs 1006 to 4094), you must first set VTP mode to transparent. Extended-range VLANs created in transparent mode are not added to the VLAN

database but are saved in the device running configuration. With VTP version 3, you can create extended-range VLANs in client or server mode. These VLANs are saved in the VLAN database.

In a switch stack, the VLAN database is downloaded to all switches in a stack, and all switches in the stack build the same VLAN database. The running configuration and the saved configuration are the same for all switches in a stack.

Add ports to a VLAN by using the **switchport** interface configuration commands:

- Identify the interface.
- For a trunk port, set trunk characteristics, and, if desired, define the VLANs to which it can belong.
- For an access port, set and define the VLAN to which it belongs.

Switch Ports

Switch ports are Layer 2-only interfaces associated with a physical port. Switch ports belong to one or more VLANs. A switch port can be an access port or a trunk port. You can configure a port as an access port or trunk port or let the Dynamic Trunking Protocol (DTP) operate on a per-port basis to set the switchport mode by negotiating with the port on the other end of the link. switch ports are used for managing the physical interface and associated Layer 2 protocols and do not handle routing or bridging.

Configure switch ports by using the **switchport** interface configuration commands.

Access Ports

An access port belongs to and carries the traffic of only one VLAN (unless it is configured as a voice VLAN port). Traffic is received and sent in native formats with no VLAN tagging. Traffic arriving on an access port is assumed to belong to the VLAN assigned to the port. If an access port receives a tagged packet (Inter-Switch Link [ISL] or IEEE 802.1Q tagged), the packet is dropped, and the source address is not learned.

The types of access ports supported are:

- Static access ports are manually assigned to a VLAN (or through a RADIUS server for use with IEEE 802.1x.

You can also configure an access port with an attached Cisco IP Phone to use one VLAN for voice traffic and another VLAN for data traffic from a device attached to the phone.

Trunk Ports

A trunk port carries the traffic of multiple VLANs and by default is a member of all VLANs in the VLAN database.

Although by default, a trunk port is a member of every VLAN known to the VTP, you can limit VLAN membership by configuring an allowed list of VLANs for each trunk port. The list of allowed VLANs does not affect any other port but the associated trunk port. By default, all possible VLANs (VLAN ID 1 to 4094) are in the allowed list. A trunk port can become a member of a VLAN only if VTP knows of the VLAN and if the VLAN is in the enabled state. If VTP learns of a new, enabled VLAN and the VLAN is in the allowed list for a trunk port, the trunk port automatically becomes a member of that VLAN and traffic is forwarded to and from the trunk port for that VLAN. If VTP learns of a new, enabled VLAN that is not in the allowed list for a trunk port, the port does not become a member of the VLAN, and no traffic for the VLAN is forwarded to or from the port.

Tunnel Ports

Tunnel ports are used in IEEE 802.1Q tunneling to segregate the traffic of customers in a service-provider network from other customers who are using the same VLAN number. You configure an asymmetric link from a tunnel port on a service-provider edge switch to an IEEE 802.1Q trunk port on the customer switch. Packets entering the tunnel port on the edge switch, already IEEE 802.1Q-tagged with the customer VLANs, are encapsulated with another layer of an IEEE 802.1Q tag (called the metro tag), containing a VLAN ID unique in the service-provider network, for each customer. The double-tagged packets go through the service-provider network keeping the original customer VLANs separate from those of other customers. At the outbound interface, also a tunnel port, the metro tag is removed, and the original VLAN numbers from the customer network are retrieved.

Tunnel ports cannot be trunk ports or access ports and must belong to a VLAN unique to each customer.

Routed Ports

A routed port is a physical port that acts like a port on a router; it does not have to be connected to a router. A routed port is not associated with a particular VLAN, as is an access port. A routed port behaves like a regular router interface, except that it does not support VLAN subinterfaces. Routed ports can be configured with a Layer 3 routing protocol. A routed port is a Layer 3 interface only and does not support Layer 2 protocols, such as DTP and STP.

Configure routed ports by putting the interface into Layer 3 mode with the **no switchport** interface configuration command. Then assign an IP address to the port, enable routing, and assign routing protocol characteristics by using the **ip routing** and **router** *protocol* global configuration commands.

Note Entering a **no switchport** interface configuration command shuts down the interface and then re-enables it, which might generate messages on the device to which the interface is connected. When you put an interface that is in Layer 2 mode into Layer 3 mode, the previous configuration information related to the affected interface might be lost.

The number of routed ports that you can configure is not limited by software. However, the interrelationship between this number and the number of other features being configured might impact CPU performance because of hardware limitations.

Note The IP Base imagesupports static routing and the Routing Information Protocol (RIP). For full Layer 3 routing or for fallback bridging, you must enable the IP Services image on the standalone device, or the active device

Switch Virtual Interfaces

A switch virtual interface (SVI) represents a VLAN of switch ports as one interface to the routing or bridging function in the system. You can associate only one SVI with a VLAN. You configure an SVI for a VLAN only to route between VLANs or to provide IP host connectivity to the device. By default, an SVI is created for the default VLAN (VLAN 1) to permit remote device administration. Additional SVIs must be explicitly configured.

Note You cannot delete interface VLAN 1.

SVIs provide IP host connectivity only to the system. SVIs are created the first time that you enter the **vlan** interface configuration command for a VLAN interface. The VLAN corresponds to the VLAN tag associated with data frames on an ISL or IEEE 802.1Q encapsulated trunk or the VLAN ID configured for an access port. Configure a VLAN interface for each VLAN for which you want to route traffic, and assign it an IP address.

You can also use the interface range command to configure existing VLAN SVIs within the range. The commands entered under the interface range command are applied to all existing VLAN SVIs within the range. You can enter the command **interface range create vlan** *x - y* to create all VLANs in the specified range that do not already exist. When the VLAN interface is created, **interface range vlan** *id* can be used to configure the VLAN interface.

Although the switch stack or device supports a total of 1005 VLANs and SVIs, the interrelationship between the number of SVIs and routed ports and the number of other features being configured might impact CPU performance because of hardware limitations.

When you create an SVI, it does not become active until it is associated with a physical port.

SVI Autostate Exclude

The line state of an SVI with multiple ports on a VLAN is in the *up* state when it meets these conditions:

- The VLAN exists and is active in the VLAN database on the device
- The VLAN interface exists and is not administratively down.
- At least one Layer 2 (access or trunk) port exists, has a link in the *up* state on this VLAN, and is in the spanning-tree forwarding state on the VLAN.

Note The protocol link state for VLAN interfaces come up when the first switchport belonging to the corresponding VLAN link comes up and is in STP forwarding state.

The default action, when a VLAN has multiple ports, is that the SVI goes down when all ports in the VLAN go down. You can use the SVI autostate exclude feature to configure a port so that it is not included in the SVI line-state up-or-down calculation. For example, if the only active port on the VLAN is a monitoring port, you might configure autostate exclude on that port so that the VLAN goes down when all other ports go down. When enabled on a port, **autostate exclude** applies to all VLANs that are enabled on that port.

The VLAN interface is brought up when one Layer 2 port in the VLAN has had time to converge (transition from STP listening-learning state to forwarding state). This prevents features such as routing protocols from using the VLAN interface as if it were fully operational and minimizes other problems, such as routing black holes.

EtherChannel Port Groups

EtherChannel port groups treat multiple switch ports as one switch port. These port groups act as a single logical port for high-bandwidth connections between devices or between devices and servers. An EtherChannel balances the traffic load across the links in the channel. If a link within the EtherChannel fails, traffic previously carried over the failed link changes to the remaining links. You can group multiple trunk ports into one logical trunk port, group multiple access ports into one logical access port, group multiple tunnel ports into one logical tunnel port, or group multiple routed ports into one logical routed port. Most protocols operate over either single ports or aggregated switch ports and do not recognize the physical ports within the port group. Exceptions

are the DTP, the Cisco Discovery Protocol (CDP), and the Port Aggregation Protocol (PAgP), which operate only on physical ports.

When you configure an EtherChannel, you create a port-channel logical interface and assign an interface to the EtherChannel. For Layer 3 interfaces, you manually create the logical interface by using the **interface port-channel** global configuration command. Then you manually assign an interface to the EtherChannel by using the **channel-group** interface configuration command. For Layer 2 interfaces, use the **channel-group** interface configuration command to dynamically create the port-channel logical interface. This command binds the physical and logical ports together.

10-Gigabit Ethernet Interfaces

A 10-Gigabit Ethernet interface operates only in full-duplex mode. The interface can be configured as a switched or routed port.

The device has a network module slot into which you can insert a 10-Gigabit Ethernet network module, a 1-Gigabit Ethernet network module, or a blank module.

For more information about the Cisco TwinGig Converter Module, see the device hardware installation guide and your transceiver module documentation.

Multigigabit Ethernet

The following Cisco switches support the mGig feature:

- WS-C3850-12X48U
- WS-C3850-24XU

Multigigabit Ethernet supports multi-rate speeds where the ports exchange auto-negotiation pages to establish a link at the highest speed that is supported by both ends of the channel. In a high-noise environment, when port speed downshifting is enabled on an interface, the line rate automatically downgrades to a lower speed when a higher speed link cannot be established or when an established link quality has degraded to a level where the PHY needs to reestablish the link. The following downshift speed values are recommended:

- 10Gbs (downshift to 5Gbs)
- 5Gbs (downshift to 2.5Gbs)
- 2.5Gbs (downshift to 1Gbs)
- 1Gbs (downshift to 100Mbs)

Power over Ethernet

The Power over Ethernet (PoE) technology allows PoE (802.3af standard), PoE+ (802.3at) ports to supply power for the operation of a device.

Cisco Universal Power Over Ethernet (Cisco UPoE) extends the IEEE PoE+ standard to double the power per port to 60 watts.

For more information, see the *Configuring PoE* section of this guide

Using the Switch USB Ports

The device has two USB ports on the front panel — a USB mini-Type B console port and a USB Type A port.

USB Mini-Type B Console Port

The device has the following console ports:

- USB mini-Type B console connection
- RJ-45 console port

Console output appears on devices connected to both ports, but console input is active on only one port at a time. By default, the USB connector takes precedence over the RJ-45 connector.

Note Windows PCs require a driver for the USB port. See the hardware installation guide for driver installation instructions.

Use the supplied USB Type A-to-USB mini-Type B cable to connect a PC or other device to the device. The connected device must include a terminal emulation application. When the device detects a valid USB connection to a powered-on device that supports host functionality (such as a PC), input from the RJ-45 console is immediately disabled, and input from the USB console is enabled. Removing the USB connection immediately reenables input from the RJ-45 console connection. An LED on the device shows which console connection is in use.

Console Port Change Logs

At software startup, a log shows whether the USB or the RJ-45 console is active. Each device in a stack issues this log. Every device always first displays the RJ-45 media type.

In the sample output, Device 1 has a connected USB console cable. Because the bootloader did not change to the USB console, the first log from Device 1 shows the RJ-45 console. A short time later, the console changes and the USB console log appears. Device 2 and Device 3 have connected RJ-45 console cables.

```
switch-stack-1
*Mar  1 00:01:00.171: %USB_CONSOLE-6-MEDIA_RJ45: Console media-type is RJ45.
*Mar  1 00:01:00.431: %USB_CONSOLE-6-MEDIA_USB: Console media-type is USB.


switch-stack-2
*Mar  1 00:01:09.835: %USB_CONSOLE-6-MEDIA_RJ45: Console media-type is RJ45.

switch-stack-3
*Mar  1 00:01:10.523: %USB_CONSOLE-6-MEDIA_RJ45: Console media-type is RJ45.
```

When the USB cable is removed or the PC de-activates the USB connection, the hardware automatically changes to the RJ-45 console interface:

```
switch-stack-1
Mar  1 00:20:48.635: %USB_CONSOLE-6-MEDIA_RJ45: Console media-type is RJ45.
```

You can configure the console type to always be RJ-45, and you can configure an inactivity timeout for the USB connector.

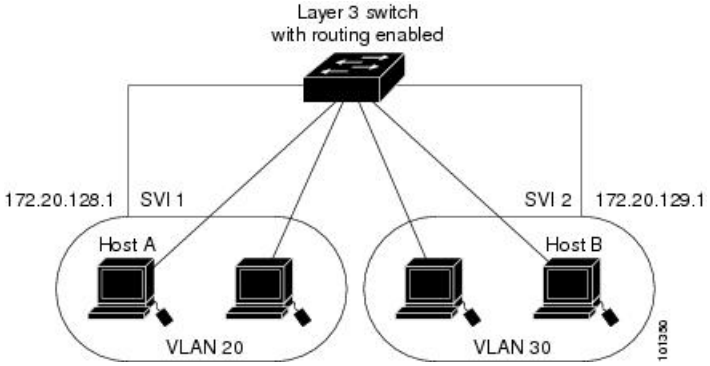
USB Type A Port

The USB Type A port provides access to external USB flash devices, also known as thumb drives or USB keys. The port supports Cisco USB flash drives with capacities from 128 MB to 8 GB (USB devices with port densities of 128 MB, 256 MB, 1 GB, 4 GB, 8 GB are supported). You can use standard Cisco IOS command- line interface (CLI) commands to read, write, erase, and copy to or from the flash device. You can also configure the device to boot from the USB flash drive.

Interface Connections

Devices within a single VLAN can communicate directly through any switch. Ports in different VLANs cannot exchange data without going through a routing device. With a standard Layer 2 device, ports in different VLANs have to exchange information through a router. By using the device with routing enabled, when you configure both VLAN 20 and VLAN 30 with an SVI to which an IP address is assigned, packets can be sent from Host A to Host B directly through the device with no need for an external router.

Figure 1: Connecting VLANs with the Switch



- The routing function can be enabled on all SVIs and routed ports. The device routes only IP traffic. When IP routing protocol parameters and address configuration are added to an SVI or routed port, any IP traffic received from these ports is routed.
- Fallback bridging forwards traffic that the device does not route or traffic belonging to a nonroutable protocol, such as DECnet. Fallback bridging connects multiple VLANs into one bridge domain by bridging between two or more SVIs or routed ports. When configuring fallback bridging, you assign SVIs or routed ports to bridge groups with each SVI or routed port assigned to only one bridge group. All interfaces in the same group belong to the same bridge domain.

Interface Configuration Mode

The device supports these interface types:

- Physical ports—device ports and routed ports
- VLANs—switch virtual interfaces
- Port channels—EtherChannel interfaces

You can also configure a range of interfaces.

To configure a physical interface (port), specify the interface type, stack member number (only stacking-capable switches), module number, and device port number, and enter interface configuration mode.

- Type—Gigabit Ethernet (gigabitethernet or gi) for 10/100/1000 Mb/s Ethernet ports, 10-Gigabit Ethernet (tengigabitethernet or te) for 10,000 Mb/s, or small form-factor pluggable (SFP) module Gigabit Ethernet interfaces (gigabitethernet or gi).
- Stack member number—The number that identifies the device within the stack. The device number range is 1 to 9 and is assigned the first time the device initializes. The default device number, before it is integrated into a device stack, is 1. When a device has been assigned a stack member number, it keeps that number until another is assigned to it.

  You can use the switch port LEDs in Stack mode to identify the stack member number of a device.
- Module number—The module or slot number on the device: switch (downlink) ports are 0, and uplink ports are 1.
- Port number—The interface number on the device. The 10/100/1000 port numbers always begin at 1, starting with the far left port when facing the front of the device, for example, gigabitethernet1/0/1 or gigabitethernet1/0/8.

  On a device with SFP uplink ports, the module number is 1 and the port numbers restart. For example, if the device has 24 10/100/1000 ports, the SFP module ports are gigabitethernet1/1/1 through gigabitethernet1/1/4 or tengigabitethernet1/1/1 through tengigabitethernet1/1/4.

You can identify physical interfaces by physically checking the interface location on the device. You can also use the **show** privileged EXEC commands to display information about a specific interface or all the interfaces on the switch. The remainder of this chapter primarily provides physical interface configuration procedures.

These are examples of how to identify interfaces on a stacking-capable device:

- To configure 10/100/1000 port 4 on a standalone device, enter this command:

```
Device(config)# interface gigabitethernet1/0/4
```

- To configure 10-Gigabit Ethernet port 1 on a standalone device, enter this command:

```
Device(config)# interface tengigabitethernet1/0/1
```

- To configure 10-Gigabit Ethernet port on stack member 3, enter this command:

```
Device(config)# interface tengigabitethernet3/0/1
```

- To configure the first SFP module (uplink) port on a standalone device, enter this command:

```
Device(config)# interface gigabitethernet1/1/1
```

Default Ethernet Interface Configuration

To configure Layer 2 parameters, if the interface is in Layer 3 mode, you must enter the **switchport** interface configuration command without any parameters to put the interface into Layer 2 mode. This shuts down the

interface and then re-enables it, which might generate messages on the device to which the interface is connected. When you put an interface that is in Layer 3 mode into Layer 2 mode, the previous configuration information related to the affected interface might be lost, and the interface is returned to its default configuration.

This table shows the Ethernet interface default configuration, including some features that apply only to Layer 2 interfaces.

Table 1: Default Layer 2 Ethernet Interface Configuration

Feature	Default Setting
Operating mode	Layer 2 or switching mode (**switchport** command).
Allowed VLAN range	VLANs 1– 4094.
Default VLAN (for access ports)	VLAN 1 (Layer 2 interfaces only).
Native VLAN (for IEEE 802.1Q trunks)	VLAN 1 (Layer 2 interfaces only).
VLAN trunking	Switchport mode dynamic auto (supports DTP) (Layer 2 interfaces only).
Port enable state	All ports are enabled.
Port description	None defined.
Speed	Autonegotiate. (Not supported on the 10-Gigabit interfaces.)
Duplex mode	Autonegotiate. (Not supported on the 10-Gigabit interfaces.)
Flow control	Flow control is set to **receive: off**. It is always off for sent packets.
EtherChannel (PAgP)	Disabled on all Ethernet ports.
Port blocking (unknown multicast and unknown unicast traffic)	Disabled (not blocked) (Layer 2 interfaces only).
Broadcast, multicast, and unicast storm control	Disabled.
Protected port	Disabled (Layer 2 interfaces only).
Port security	Disabled (Layer 2 interfaces only).
Port Fast	Disabled.
Auto-MDIX	Enabled. **Note** The switch might not support a pre-standard powered device—such as Cisco IP phones and access points that do not fully support IEEE 802.3af—if that powered device is connected to the switch through a crossover cable. This is regardless of whether auto-MIDX is enabled on the switch port.

Feature	Default Setting
Power over Ethernet (PoE)	Enabled (auto).

Interface Speed and Duplex Mode

Ethernet interfaces on the switch operate at 10, 100, 1000, or 10,000 Mb/s and in either full- or half-duplex mode. In full-duplex mode, two stations can send and receive traffic at the same time. Normally, 10-Mb/s ports operate in half-duplex mode, which means that stations can either receive or send traffic.

Switch modules include Gigabit Ethernet (10/100/1000-Mb/s) ports, 10-Gigabit Ethernet ports, and small form-factor pluggable (SFP) module slots supporting SFP modules.

Speed and Duplex Configuration Guidelines

When configuring an interface speed and duplex mode, note these guidelines:

- The 10-Gigabit Ethernet ports do not support the speed and duplex features. These ports operate only at 10,000 Mb/s and in full-duplex mode.
- Do not disable Auto-Negotiation on PoE switches.
- Gigabit Ethernet (10/100/1000-Mb/s) ports support all speed options and all duplex options (auto, half, and full). However, Gigabit Ethernet ports operating at 1000 Mb/s do not support half-duplex mode.
- For SFP module ports, the speed and duplex CLI options change depending on the SFP module type:
  - The 1000BASE-*x* (where -*x* is -BX, -CWDM, -LX, -SX, and -ZX) SFP module ports support the **nonegotiate** keyword in the **speed** interface configuration command. Duplex options are not supported.
  - The 1000BASE-T SFP module ports support the same speed and duplex options as the 10/100/1000-Mb/s ports.
- If both ends of the line support autonegotiation, we highly recommend the default setting of **auto** negotiation.
- If one interface supports autonegotiation and the other end does not, configure duplex and speed on both interfaces; do not use the **auto** setting on the supported side.
- When STP is enabled and a port is reconfigured, the device can take up to 30 seconds to check for loops. The port LED is amber while STP reconfigures.
- As best practice, we suggest configuring the speed and duplex options on a link to auto or to fixed on both the ends. If one side of the link is configured to auto and the other side is configured to fixed, the link will not be up and this is expected.

  The speed on the switch is autonegotiable and it can detect a peer device only if the peer device supports autonegotiation. If the peer device does not support autonegotiation, and autonegotiation is enabled on the switch, the switch goes into half duplex mode.

Caution Changing the interface speed and duplex mode configuration might shut down and re-enable the interface during the reconfiguration.

IEEE 802.3x Flow Control

Flow control enables connected Ethernet ports to control traffic rates during congestion by allowing congested nodes to pause link operation at the other end. If one port experiences congestion and cannot receive any more traffic, it notifies the other port by sending a pause frame to stop sending until the condition clears. Upon receipt of a pause frame, the sending device stops sending any data packets, which prevents any loss of data packets during the congestion period.

Note Flow control is not supported on Cisco Catalyst 3650 Series Switches and Cisco Catalyst 3850 Series Switches.

Note The switch ports can receive, but not send, pause frames.

You use the **flowcontrol** interface configuration command to set the interface's ability to **receive** pause frames to **on**, **off**, or **desired**. The default state is **on**.

When set to **desired**, an interface can operate with an attached device that is required to send flow-control packets or with an attached device that is not required to but can send flow-control packets.

These rules apply to flow control settings on the device:

- **receive on** (or **desired**): The port cannot send pause frames but can operate with an attached device that is required to or can send pause frames; the port can receive pause frames.
- **receive off**: Flow control does not operate in either direction. In case of congestion, no indication is given to the link partner, and no pause frames are sent or received by either device.

Layer 3 Interfaces

The device supports these types of Layer 3 interfaces:

- SVIs: You should configure SVIs for any VLANs for which you want to route traffic. SVIs are created when you enter a VLAN ID following the **interface vlan** global configuration command. To delete an SVI, use the **no interface vlan** global configuration command. You cannot delete interface VLAN 1.

  Note When you create an SVI, it does not become active until it is associated with a physical port.

  When configuring SVIs, you can also configure SVI autostate exclude on a port in the SVI to exclude that port from being included in determining SVI line-state status.

- Routed ports: Routed ports are physical ports configured to be in Layer 3 mode by using the **no switchport** interface configuration command.
- Layer 3 EtherChannel ports: EtherChannel interfaces made up of routed ports.

A Layer 3 device can have an IP address assigned to each routed port and SVI.

There is no defined limit to the number of SVIs and routed ports that can be configured in a device or in a device stack. However, the interrelationship between the number of SVIs and routed ports and the number of other features being configured might have an impact on CPU usage because of hardware limitations. If the device is using its maximum hardware resources, attempts to create a routed port or SVI have these results:

- If you try to create a new routed port, the device generates a message that there are not enough resources to convert the interface to a routed port, and the interface remains as a switchport.
- If you try to create an extended-range VLAN, an error message is generated, and the extended-range VLAN is rejected.
- If the device is notified by VLAN Trunking Protocol (VTP) of a new VLAN, it sends a message that there are not enough hardware resources available and shuts down the VLAN. The output of the **show vlan** user EXEC command shows the VLAN in a suspended state.
- If the device attempts to boot up with a configuration that has more VLANs and routed ports than hardware can support, the VLANs are created, but the routed ports are shut down, and the device sends a message that this was due to insufficient hardware resources.

Note All Layer 3 interfaces require an IP address to route traffic. This procedure shows how to configure an interface as a Layer 3 interface and how to assign an IP address to an interface:

If the physical port is in Layer 2 mode (the default), you must enter the **no switchport** interface configuration command to put the interface into Layer 3 mode. Entering a **no switchport** command disables and then re-enables the interface, which might generate messages on the device to which the interface is connected. Furthermore, when you put an interface that is in Layer 2 mode into Layer 3 mode, the previous configuration information related to the affected interface might be lost, and the interface is returned to its default configuration

How to Configure Interface Characteristics

Configuring Interfaces

These general instructions apply to all interface configuration processes.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:**	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
	`Device> enable`	
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.

Step 3	**interface**
	Example:
	`Device(config)# interface gigabitethernet 1/0/1`
	`Device(config-if)#`
		Identifies the interface type, the device number (only on stacking-capable switches), and the number of the connector.
		Note You do not need to add a space between the interface type and the interface number. For example, in the preceding line, you can specify either **gigabitethernet 1/0/1**, **gigabitethernet1/0/1**, **gi 1/0/1**, or **gi1/0/1**.

Step 4	Follow each **interface** command with the interface configuration commands that the interface requires.	Defines the protocols and applications that will run on the interface. The commands are collected and applied to the interface when you enter another interface command or enter **end** to return to privileged EXEC mode.
Step 5	**interface range** or **interface range macro**	(Optional) Configures a range of interfaces. **Note** Interfaces configured in a range must be the same type and must be configured with the same feature options.
Step 6	**show interfaces**	Displays a list of all interfaces on or configured for the switch. A report is provided for each interface that the device supports or for the specified interface.

Adding a Description for an Interface

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:**	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
	`Device> enable`	
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**interface** *interface-id* **Example:** `Device(config)# interface gigabitethernet 1/0/2`	Specifies the interface for which you are adding a description, and enter interface configuration mode.
Step 4	**description** *string* **Example:** `Device(config-if)# description Connects to Marketing`	Adds a description (up to 240 characters) for an interface.
Step 5	**end** **Example:** `Device(config-if)# end`	Returns to privileged EXEC mode.
Step 6	**show interfaces** *interface-id* **description**	Verifies your entry.
Step 7	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring a Range of Interfaces

To configure multiple interfaces with the same configuration parameters, use the **interface range** global configuration command. When you enter the interface-range configuration mode, all command parameters that you enter are attributed to all interfaces within that range until you exit this mode.

Procedure

	Command or Action	Purpose
Step 1	**enable**	Enables privileged EXEC mode.

	Command or Action	Purpose
	Example: `Device> enable`	• Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**interface range** {*port-range* | **macro** *macro_name*} **Example:** `Device(config)# interface range macro`	Specifies the range of interfaces (VLANs or physical ports) to be configured, and enter interface-range configuration mode. • You can use the **interface range** command to configure up to five port ranges or a previously defined macro. • The **macro** variable is explained in the section on *Configuring and Using Interface Range Macros.* • In a comma-separated *port-range*, you must enter the interface type for each entry and enter spaces before and after the comma. • In a hyphen-separated *port-range*, you do not need to re-enter the interface type, but you must enter a space before the hyphen. **Note** Use the normal configuration commands to apply the configuration parameters to all interfaces in the range. Each command is executed as it is entered.
Step 4	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.
Step 5	**show interfaces** [*interface-id*] **Example:** `Device# show interfaces`	Verifies the configuration of the interfaces in the range.

	Command or Action	Purpose
Step 6	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring and Using Interface Range Macros

You can create an interface range macro to automatically select a range of interfaces for configuration. Before you can use the **macro** keyword in the **interface range macro** global configuration command string, you must use the **define interface-range** global configuration command to define the macro.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**define interface-range** *macro_name interface-range* **Example:** `Device(config)# define interface-range enet_list gigabitethernet 1/0/1 - 2`	Defines the interface-range macro, and save it in NVRAM. • The *macro_name* is a 32-character maximum character string. • A macro can contain up to five comma-separated interface ranges. • Each *interface-range* must consist of the same port type. **Note** Before you can use the **macro** keyword in the **interface range macro** global configuration command string, you must use the **define interface-range** global configuration command to define the macro.

	Command or Action	Purpose
Step 4	**interface range macro** *macro_name* **Example:** `Device(config)# interface range macro enet_list`	Selects the interface range to be configured using the values saved in the interface-range macro called *macro_name*. You can now use the normal configuration commands to apply the configuration to all interfaces in the defined macro.
Step 5	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.
Step 6	**show running-config | include define** **Example:** `Device# show running-config | include define`	Shows the defined interface range macro configuration.
Step 7	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring Multigigabit Ethernet Parameters

Procedure

	Command or Action	Purpose
Step 1	**interface tengigabitethernet** *interface number* **Example:** `Device(config)#interface tengigabitethernet 1/1/37`	Configures the 10 Gigabit Ethernet interface.
Step 2	**speed auto** **Example:** `Device(config-if)#speed auto`	Sets the speed to auto speed negotiation.
Step 3	**downshift** **Example:**	Enables downshift on the specified interface. When downshift is enabled, the port speed gets downshifted or lowered, if the link quality is bad or if the link is continuously down.

	Command or Action	Purpose
	`Device(config-if)#downshift`	
Step 4	**no downshift** **Example:** `Device(config-if)#no downshift`	Disables downshift on the specified interface. By default, downshift is enabled on all the multigigabit ports. Use the **no downshift** command to disable downshift on an interface.
Step 5	**end** **Example:** `Device(config-if)#end`	Returns to privileged EXEC mode.
Step 6	**show interfaces downshift** **Example:** `Device# show interfaces downshift`	(Optional) Displays downshift status of all the multigigabit ports.
Step 7	**show intefaces** *interface-number* **downshift** **Example:** `Device# show interfaces TenGigabitEthernet 1/0/1 downshift`	(Optional) Displays downshift status of the specified multigigabit port.
Step 8	**show intefaces downshift module** *module-number* **Example:** `Device# show interface downshift module 1`	(Optional) Displays downshift status of the specified module.

Configuring IEEE 802.3x Flow Control

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode
Step 2	**interface** *interface-id* **Example:** `Device(config)# interface gigabitethernet`	Specifies the physical interface to be configured, and enter interface configuration mode.

	Command or Action	Purpose
	`1/0/1`	
Step 3	**flowcontrol** {**receive**} {**on** | **off** | **desired**} **Example:** `Device(config-if)# flowcontrol receive on`	Configures the flow control mode for the port.
Step 4	**end** **Example:** `Device(config-if)# end`	Returns to privileged EXEC mode.
Step 5	**show interfaces** *interface-id* **Example:** `Device# show interfaces gigabitethernet 1/0/1`	Verifies the interface flow control settings.
Step 6	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring Layer 3 Interfaces

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**interface** {**gigabitethernet** *interface-id*} | {**vlan** *vlan-id*} | {**port-channel** *port-channel-number*} **Example:** `Device(config)# interface gigabitethernet1/0/2`	Specifies the interface to be configured as a Layer 3 interface, and enter interface configuration mode.
Step 4	**no switchport** **Example:** `Device(config-if)# no switchport`	For physical ports only, enters Layer 3 mode.
Step 5	**ip address** *ip_address subnet_mask* **Example:** `Device(config-if)# ip address 192.20.135.21 255.255.255.0`	Configures the IP address and IP subnet.
Step 6	**no shutdown** **Example:** `Device(config-if)# no shutdown`	Enables the interface.
Step 7	**end** **Example:** `Device(config-if)# end`	Returns to privileged EXEC mode.
Step 8	**show interfaces** [*interface-id*]	Verifies the configuration.
Step 9	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring Logical Layer 3 GRE Tunnel Interfaces

Before you begin

Generic Routing Encapsulation (GRE) is a tunneling protocol used to encapsulate network layer protocols inside virtual point-to-point links. A GRE tunnel only provides encapsulation and not encryption.

Attention Beginning in Cisco IOS XE Release 3.7.2E, GRE tunnels are supported on the hardware on Cisco Catalyst switches. When GRE is configured without tunnel options, packets are hardware-switched. When GRE is configured with tunnel options (such as key, checksum, etc.), packets are switched in the software. A maximum of 10 GRE tunnels are supported.

Note Other features like Access Control Lists (ACL) and Quality of Service (QoS) are not supported for the GRE tunnels.

To configure a GRE tunnel, perform this task:

Procedure

	Command or Action	Purpose
Step 1	**interface tunnel** *number* **Example:** `Device(config)#interface tunnel 2`	Enables tunneling on the interface.
Step 2	**ip address***ip_addresssubnet_mask* **Example:** `Device(config)#ip address 100.1.1.1 255.255.255.0`	Configures the IP address and IP subnet.
Step 3	**tunnel source**{*ip_address* | *type_number*} **Example:** `Device(config)#tunnel source 10.10.10.1`	Configures the tunnel source.
Step 4	**tunnel destination**{*host_name* | *ip_address*} **Example:** `Device(config)#tunnel destination 10.10.10.2`	Configures the tunnel destination.
Step 5	**tunnel mode gre ip** **Example:** `Device(config)#tunnel mode gre ip`	Configures the tunnel mode.
Step 6	**end** **Example:** `Device(config)#end`	Exist configuration mode.

Configuring SVI Autostate Exclude

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**interface** *interface-id* **Example:** `Device(config)# interface gigabitethernet1/0/2`	Specifies a Layer 2 interface (physical port or port channel), and enter interface configuration mode.
Step 4	**switchport autostate exclude** **Example:** `Device(config-if)# switchport autostate exclude`	Excludes the access or trunk port when defining the status of an SVI line state (up or down)
Step 5	**end** **Example:** `Device(config-if)# end`	Returns to privileged EXEC mode.
Step 6	**show running config interface** *interface-id*	(Optional) Shows the running configuration. Verifies the configuration.
Step 7	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Shutting Down and Restarting the Interface

Shutting down an interface disables all functions on the specified interface and marks the interface as unavailable on all monitoring command displays. This information is communicated to other network servers through all dynamic routing protocols. The interface is not mentioned in any routing updates.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**interface** {**vlan** *vlan-id*} | {**gigabitethernet***interface-id*} | {**port-channel** *port-channel-number*} **Example:** `Device(config)# interface gigabitethernet 1/0/2`	Selects the interface to be configured.
Step 4	**shutdown** **Example:** `Device(config-if)# shutdown`	Shuts down an interface.
Step 5	**no shutdown** **Example:** `Device(config-if)# no shutdown`	Restarts an interface.
Step 6	**end** **Example:** `Device(config-if)# end`	Returns to privileged EXEC mode.
Step 7	**show running-config** **Example:**	Verifies your entries.

	Command or Action	Purpose
	`Device# show running-config`	

Configuring the Console Media Type

Follow these steps to set the console media type to RJ-45. If you configure the console as RJ-45, USB console operation is disabled, and input comes only through the RJ-45 connector.

This configuration applies to all switches in a stack.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**line console 0** **Example:** `Device(config)# line console 0`	Configures the console and enters line configuration mode.
Step 4	**media-type rj45** **Example:** `Device(config-line)# media-type rj45`	Configures the console media type to be only RJ-45 port. If you do not enter this command and both types are connected, the USB port is used by default.
Step 5	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.
Step 6	**copy running-config startup-config** **Example:** `Device# copy running-config`	(Optional) Saves your entries in the configuration file.

	Command or Action	Purpose
	`startup-config`	

Configuring the USB Inactivity Timeout

The configurable inactivity timeout reactivates the RJ-45 console port if the USB console port is activated but no input activity occurs on it for a specified time period. When the USB console port is deactivated due to a timeout, you can restore its operation by disconnecting and reconnecting the USB cable.

Note The configured inactivity timeout applies to all devices in a stack. However, a timeout on one device does not cause a timeout on other devices in the stack.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**line console 0** **Example:** `Device(config)# line console 0`	Configures the console and enters line configuration mode.
Step 4	**usb-inactivity-timeout** *timeout-minutes* **Example:** `Device(config-line)# usb-inactivity-timeout 30`	Specify an inactivity timeout for the console port. The range is 1 to 240 minutes. The default is to have no timeout configured.
Step 5	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Monitoring Interface Characteristics

Monitoring Interface Status

Commands entered at the privileged EXEC prompt display information about the interface, including the versions of the software and the hardware, the configuration, and statistics about the interfaces.

Table 2: Show Commands for Interfaces

Command	Purpose
show interfaces *interface-number* **downshift module***module-number*	Displays the downshift status details of the specified interfaces and modules.
show interfaces *interface-id* **status** [**err-disabled**]	Displays interface status or a list of interfaces in the error-disabled state.
show interfaces [*interface-id*] **switchport**	Displays administrative and operational status of switching (nonrouting) ports. You can use this command to find out if a port is in routing or in switching mode.
show interfaces [*interface-id*] **description**	Displays the description configured on an interface or all interfaces and the interface status.
show ip interface [*interface-id*]	Displays the usability status of all interfaces configured for IP routing or the specified interface.
show interface [*interface-id*] **stats**	Displays the input and output packets by the switching path for the interface.
show interfaces *interface-id*	(Optional) Displays speed and duplex on the interface.
show interfaces transceiver dom-supported-list	(Optional) Displays Digital Optical Monitoring (DOM) status on the connect SFP modules.
show interfaces transceiver properties	(Optional) Displays temperature, voltage, or amount of current on the interface.
show interfaces [*interface-id*] [{**transceiver properties** | **detail**}] *module number*]	Displays physical and operational status about an SFP module.
show running-config interface [*interface-id*]	Displays the running configuration in RAM for the interface.
show version	Displays the hardware configuration, software version, the names and sources of configuration files, and the boot images.
show controllers ethernet-controller *interface-id* **phy**	Displays the operational state of the auto-MDIX feature on the interface.

Clearing and Resetting Interfaces and Counters

Table 3: Clear Commands for Interfaces

Command	Purpose
clear counters [*interface-id*]	Clears interface counters.
clear interface *interface-id*	Resets the hardware logic on an interface.
clear line [*number* | **console 0** | **vty** *number*]	Resets the hardware logic on an asynchronous serial line.

Note The **clear counters** privileged EXEC command does not clear counters retrieved by using Simple Network Management Protocol (SNMP), but only those seen with the **show interface** privileged EXEC command.

Configuration Examples for Interface Characteristics

Adding a Description to an Interface: Example

Displaying Downshift Status of Interfaces: Examples

This example shows how to display the downshift status of all the multi-gigabit ports.

```
Device# show interfaces downshift

Port         Enabled         Active         AdminSpeed        OperSpeed
Te2/0/37       yes             no             auto              auto
Te2/0/38       yes             no             auto              10G
Te2/0/39       yes             no             auto              auto
Te2/0/40       yes             no             auto              10G
Te2/0/41       yes             no             auto              auto
Te2/0/42       yes             no             auto              auto
Te2/0/43       yes             yes            auto              5000
Te2/0/44       yes             no             auto              auto
Te2/0/45       yes             yes            auto              2500
Te2/0/46       yes             no             auto              auto
Te2/0/47       yes             no             auto              10G
Te2/0/48       yes             no             auto              auto
```

This example shows how to display the downshift status of the specified multi-gigabit port.

```
Device# show interfaces  te2/0/43 downshift

Port         Enabled         Active             AdminSpeed        OperSpeed
Te2/0/43       yes             yes                10G               5000
```

The fields in command output are explained below:

Port	Displays the interface number
Enabled	Indicates that Downshift is enabled (yes) / disabled (no) on the specified port
Active	Displays whether Downshift has occurred on the interface or not
AdminSpeed	Displays the speed set by the user (or) default interface speed
OperSpeed	Displays current operational speed on the interface

Identifying Interfaces on a Stack-Capable Switch: Examples

To configure 10/100/1000 port 4 on a standalone switch, enter this command:

```
Device(config)# interface gigabitethernet1/1/4
```

To configure 10-Gigabit Ethernet port 1 on a standalone switch, enter this command:

```
Device(config)# interface tengigabitethernet1/0/1
```

To configure 10-Gigabit Ethernet port on stack member 3, enter this command:

```
Device(config)# interface tengigabitethernet3/0/1
```

To configure the first SFP module uplink port on stack member 1, enter this command:

```
Device(config)# interface gigabitethernet1/1/1
```

Configuring a Range of Interfaces: Examples

This example shows how to use the **interface range** global configuration command to set the speed to 100 Mb/s on ports 1 to 4 on switch 1:

```
Device# configure terminal
Device(config)# interface range gigabitethernet 1/0/1 - 4
Device(config-if-range)# speed 100
```

This example shows how to use a comma to add different interface type strings to the range to enable Gigabit Ethernet ports 1 to 3 and 10-Gigabit Ethernet ports 1 and 2 to receive flow-control pause frames:

```
Device# configure terminal
Device(config)# interface range gigabitethernet1/0/1 - 3 , tengigabitethernet1/1/1 - 2
Device(config-if-range)# flowcontrol receive on
```

If you enter multiple configuration commands while you are in interface-range mode, each command is executed as it is entered. The commands are not batched and executed after you exit interface-range mode. If you exit interface-range configuration mode while the commands are being executed, some commands might

not be executed on all interfaces in the range. Wait until the command prompt reappears before exiting interface-range configuration mode.

Configuring and Using Interface Range Macros: Examples

This example shows how to define an interface-range named *enet_list* to include ports 1 and 2 on switch 1 and to verify the macro configuration:

```
Device# configure terminal
Device(config)# define interface-range enet_list gigabitethernet 1/1/1 - 2
Device(config)# end
Device# show running-config | include define
define interface-range enet_list gigabitethernet 1/1/1 - 2
```

This example shows how to create a multiple-interface macro named *macro1*:

```
Device# configure terminal
Device(config)# define interface-range macro1 gigabitethernet1/1/1 - 2, gigabitethernet1/1/5
 - 7, tengigabitethernet1/1/1 -2
Device(config)# end
```

This example shows how to enter interface-range configuration mode for the interface-range macro *enet_list*:

```
Device# configure terminal
Device(config)# interface range macro enet_list
Device(config-if-range)#
```

This example shows how to delete the interface-range macro *enet_list* and to verify that it was deleted.

```
Device# configure terminal
Device(config)# no define interface-range enet_list
Device(config)# end
Device# show run | include define
Device#
```

Setting Interface Speed and Duplex Mode: Example

Configuring Layer 3 Interfaces: Example

Configuring the Console Media Type: Example

This example disables the USB console media type and enables the RJ-45 console media type.

```
Device# configure terminal
Device(config)# line console 0
Device(config-line)# media-type rj45
```

This configuration terminates any active USB console media type in the stack. A log shows that this termination has occurred. This example shows that the console on switch 1 reverted to RJ-45.

```
*Mar  1 00:25:36.860: %USB_CONSOLE-6-CONFIG_DISABLE: Console media-type USB disabled by
system configuration, media-type reverted to RJ45.
```

At this point no switches in the stack allow a USB console to have input. A log entry shows when a console cable is attached. If a USB console cable is connected to switch 2, it is prevented from providing input.

```
*Mar  1 00:34:27.498: %USB_CONSOLE-6-CONFIG_DISALLOW: Console media-type USB is disallowed
 by system configuration, media-type remains RJ45. (switch-stk-2)
```

This example reverses the previous configuration and immediately activates any USB console that is connected.

```
Device# configure terminal
Device(config)# line console 0
Device(config-line)# no media-type rj45
```

Configuring the USB Inactivity Timeout: Example

This example configures the inactivity timeout to 30 minutes:

```
Device# configure terminal
Device(config)# line console 0
Device(config-line)# usb-inactivity-timeout 30
```

To disable the configuration, use these commands:

```
Device# configure terminal
Device(config)# line console 0
Device(config-line)# no usb-inactivity-timeout
```

If there is no (input) activity on a USB console port for the configured number of minutes, the inactivity timeout setting applies to the RJ-45 port, and a log shows this occurrence:

```
*Mar  1 00:47:25.625: %USB_CONSOLE-6-INACTIVITY_DISABLE: Console media-type USB disabled
due to inactivity, media-type reverted to RJ45.
```

At this point, the only way to reactivate the USB console port is to disconnect and reconnect the cable.

When the USB cable on the switch has been disconnected and reconnected, a log similar to this appears:

```
*Mar  1 00:48:28.640: %USB_CONSOLE-6-MEDIA_USB: Console media-type is USB.
```

Additional References for the Interface Characteristics Feature

Related Documents

Related Topic	Document Title
Platform-independent command reference	*Interface and Hardware Command Reference, Cisco IOS XE Release 3.2SE (Catalyst 3850 Switches)*

Related Topic	Document Title
Platform-independent configuration information	*Interface and Hardware Component Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)*

Standards and RFCs

Standard/RFC	Title
None	--

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

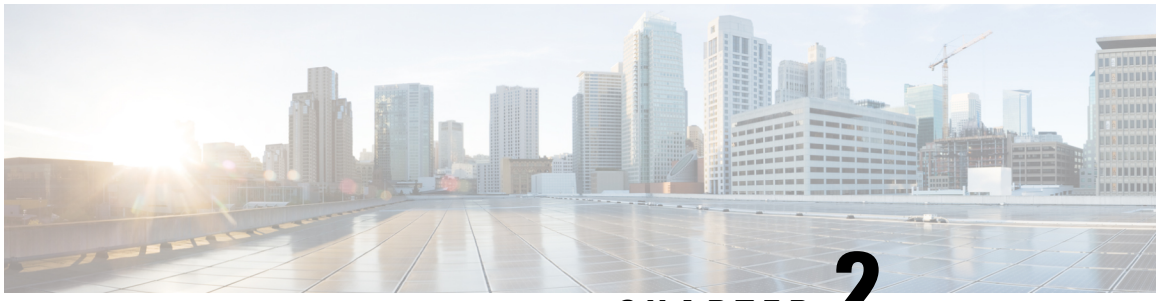
Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information for Configuring Interface Characteristics

Release	Modification
Cisco IOS XE 3.2SE	This feature was introduced.
Cisco IOS XE 3.7.2E	Support for configuring GRE tunnels in the hardware. When GRE is configured without tunnel options, packets are hardware-switched.

Release	Modification
Cisco IOS XE Denali 16.3.2	Support for downshift on mGig interfaces was introduced. When port speed downshifting is enabled on an interface, the line rate automatically downgrades to a lower speed if the link quality is bad or if the link is continuously down.
Cisco IOS XE Denali 16.3.6	Support for Digital Optical Monitoring was introduced. It enables you to monitor optical input and output power, temperature, and voltage. The feature is supported on all transceivers that support DOM and is disabled by default.



CHAPTER 2

Configuring Auto-MDIX

- Prerequisites for Auto-MDIX, on page 33
- Restrictions for Auto-MDIX, on page 33
- Information About Configuring Auto-MDIX, on page 34
- How to Configure Auto-MDIX, on page 34
- Example for Configuring Auto-MDIX, on page 35
- Additional References for Auto-MDIX, on page 36
- Feature History and Information for Auto-MDIX, on page 36

Prerequisites for Auto-MDIX

To configure Layer 2 parameters, if the interface is in Layer 3 mode, you must enter the **switchport** interface configuration command without any parameters to put the interface into Layer 2 mode. This shuts down the interface and then re-enables it, which might generate messages on the device to which the interface is connected. When you put an interface that is in Layer 3 mode into Layer 2 mode, the previous configuration information related to the affected interface might be lost, and the interface is returned to its default configuration.

Automatic medium-dependent interface crossover (auto-MDIX) is enabled by default.

Auto-MDIX is supported on all 10/100/1000-Mb/s and on 10/100/1000BASE-TX small form-factor pluggable (SFP)-module interfaces. It is not supported on 1000BASE-SX or -LX SFP module interfaces.

Restrictions for Auto-MDIX

The device might not support a pre-standard powered device—such as Cisco IP phones and access points that do not fully support IEEE 802.3af—if that powered device is connected to the device through a crossover cable. This is regardless of whether auto-MIDX is enabled on the switch port.

Information About Configuring Auto-MDIX

Auto-MDIX on an Interface

When automatic medium-dependent interface crossover (auto-MDIX) is enabled on an interface, the interface automatically detects the required cable connection type (straight through or crossover) and configures the connection appropriately. When connecting devices without the auto-MDIX feature, you must use straight-through cables to connect to devices such as servers, workstations, or routers and crossover cables to connect to other devices or repeaters. With auto-MDIX enabled, you can use either type of cable to connect to other devices, and the interface automatically corrects for any incorrect cabling. For more information about cabling requirements, see the hardware installation guide.

This table shows the link states that result from auto-MDIX settings and correct and incorrect cabling.

Table 4: Link Conditions and Auto-MDIX Settings

Local Side Auto-MDIX	Remote Side Auto-MDIX	With Correct Cabling	With Incorrect Cabling
On	On	Link up	Link up
On	Off	Link up	Link up
Off	On	Link up	Link up
Off	Off	Link up	Link down

How to Configure Auto-MDIX

Configuring Auto-MDIX on an Interface

Auto MDIX is turned on by default. To disable Auto MDIX on a port, use the **no mdix auto** command under the interface configuration mode. To put it back to default, use the **mdix auto** command in the interface configuration mode. The following steps show how to enable the Auto MDIX.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:**	Enters global configuration mode

	Command or Action	Purpose
	`Device# configure terminal`	
Step 3	**interface** *interface-id* **Example:** `Device(config)# interface gigabitethernet1/0/1`	Specifies the physical interface to be configured, and enter interface configuration mode.
Step 4	**mdix auto** **Example:** `Device(config-if)# mdix auto`	Enables the Auto MDIX feature.
Step 5	**end** **Example:** `Device(config-if)# end`	Returns to privileged EXEC mode.
Step 6	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Example for Configuring Auto-MDIX

This example shows how to enable auto-MDIX on a port:

```
Device# configure terminal
Device(config)# interface gigabitethernet1/0/1
Device(config-if)# mdix auto
Device(config-if)# end
```

Additional References for Auto-MDIX

MIBs

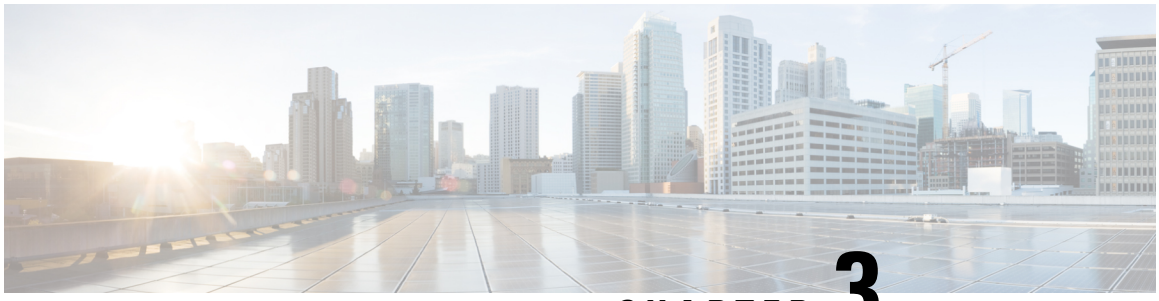
MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information for Auto-MDIX

Release	Modification
Cisco IOS XE 3.2SE	This feature was introduced.



CHAPTER 3

Configuring Ethernet Management Port

- Prerequisites for Ethernet Management Ports, on page 37
- Information About the Ethernet Management Port, on page 37
- How to Configure the Ethernet Management Port, on page 40
- Additional References for Ethernet Management Ports, on page 41
- Feature History and Information for Ethernet Management Ports, on page 41

Prerequisites for Ethernet Management Ports

When connecting a PC to the Ethernet management port, you must first assign an IP address.

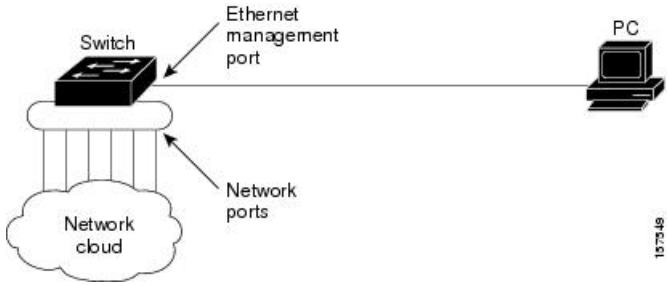
Information About the Ethernet Management Port

The Ethernet management port, also referred to as the *Gi0/0* or *GigabitEthernet0/0* port, is a VRF (VPN routing/forwarding) interface to which you can connect a PC. You can use the Ethernet management port instead of the device console port for network management. When managing a device stack, connect the PC to the Ethernet management port on a stack member.

Ethernet Management Port Direct Connection to a Device

Figure 2: Connecting a Switch to a PC

This figure displays how to connect the Ethernet management port to the PC for a device or a standalone device.

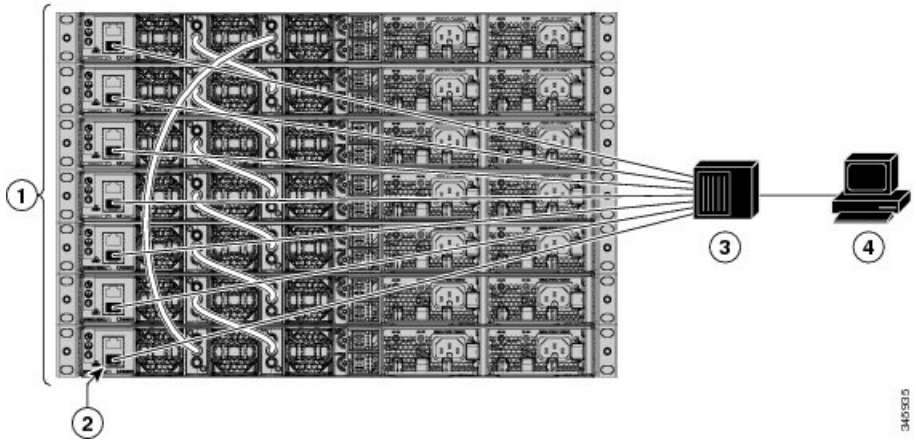


Ethernet Management Port Connection to Stack Devices using a Hub

In a stack with only stack devices, all the Ethernet management ports on the stack members are connected to a hub to which the PC is connected. The active link is from the Ethernet management port on the active switchthrough the hub, to the PC. If the activedevice fails and a new active device is elected, the active link is now from the Ethernet management port on the new active device to the PC.

Figure 3: Connecting a Device Stack to a PC

This figure displays how a PC uses a hub to connect to a device stack.



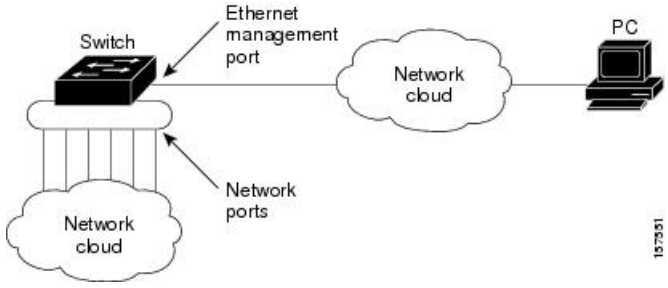
1	Switch stack	3	Hub
2	Management port	4	PC

Ethernet Management Port and Routing

By default, the Ethernet management port is enabled. The device cannot route packets from the Ethernet management port to a network port, and the reverse. Even though the Ethernet management port does not support routing, you may need to enable routing protocols on the port.

Figure 4: Network Example with Routing Protocols Enabled

Enable routing protocols on the Ethernet management port when the PC is multiple hops away from the device and the packets must pass through multiple Layer 3 devices to reach the PC.



In the above figure , if the Ethernet management port and the network ports are associated with the same routing process, the routes are propagated as follows:

- The routes from the Ethernet management port are propagated through the network ports to the network.

- The routes from the network ports are propagated through the Ethernet management port to the network.

Because routing is not supported between the Ethernet management port and the network ports, traffic between these ports cannot be sent or received. If this happens, data packet loops occur between the ports, which disrupt the device and network operation. To prevent the loops, configure route filters to avoid routes between the Ethernet management port and the network ports.

Supported Features on the Ethernet Management Port

The Ethernet management port supports these features:

- Express Setup (only in switch stacks)
- Network Assistant
- Telnet with passwords
- TFTP
- Secure Shell (SSH)
- DHCP-based autoconfiguration
- SMNP (only the ENTITY-MIB and the IF-MIB)
- IP ping
- Interface features
  - Speed—10 Mb/s, 100 Mb/s, and autonegotiation
  - Duplex mode—Full, half, and autonegotiation
  - Loopback detection
- Cisco Discovery Protocol (CDP)
- DHCP relay agent
- IPv4 and IPv6 access control lists (ACLs)
- Routing protocols

Caution Before enabling a feature on the Ethernet management port, make sure that the feature is supported. If you try to configure an unsupported feature on the Ethernet Management port, the feature might not work properly, and the device might fail.

How to Configure the Ethernet Management Port

Disabling and Enabling the Ethernet Management Port

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**interface gigabitethernet0/0** **Example:** `Device(config)# interface gigabitethernet0/0`	Specifies the Ethernet management port in the CLI.
Step 3	**shutdown** **Example:** `Device(config-if)# shutdown`	Disables the Ethernet management port.
Step 4	**no shutdown** **Example:** `Device(config-if)# no shutdown`	Enables the Ethernet management port.
Step 5	**exit** **Example:** `Device(config-if)# exit`	Exits interface configuration mode.
Step 6	**show interfaces gigabitethernet0/0** **Example:** `Device# show interfaces gigabitethernet0/0`	Displays the link status. To find out the link status to the PC, you can monitor the LED for the Ethernet management port. The LED is green (on) when the link is active, and the LED is off when the link is down. The LED is amber when there is a POST failure.

What to do next

Proceed to manage or configure your switch using the Ethernet management port. Refer to the *Network Management Configuration Guide (Catalyst 3850 Switches)*.

Additional References for Ethernet Management Ports

Related Documents

Related Topic	Document Title
Bootloader configuration	*System Management Configuration Guide (Catalyst 3850 Switches)*
Bootloader commands	*System Management Command Reference (Catalyst 3850 Switches)*

MIBs

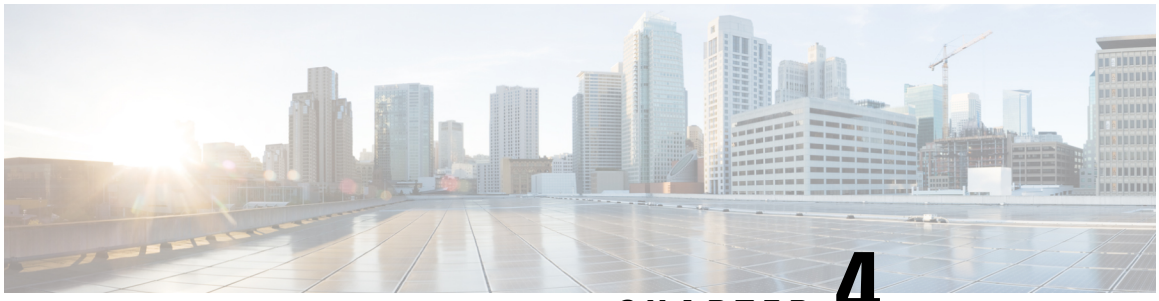
MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information for Ethernet Management Ports

Release	Modification
Cisco IOS XE 3.2SE	This feature was introduced.



CHAPTER 4

Configuring LLDP, LLDP-MED, and Wired Location Service

- Restrictions for LLDP, on page 43
- Information About LLDP, LLDP-MED, and Wired Location Service, on page 43
- How to Configure LLDP, LLDP-MED, and Wired Location Service, on page 47
- Configuration Examples for LLDP, LLDP-MED, and Wired Location Service, on page 57
- Monitoring and Maintaining LLDP, LLDP-MED, and Wired Location Service, on page 58
- Additional References for LLDP, LLDP-MED, and Wired Location Service, on page 59
- Feature Information for LLDP, LLDP-MED, and Wired Location Service, on page 59

Restrictions for LLDP

- If the interface is configured as a tunnel port, LLDP is automatically disabled.
- If you first configure a network-policy profile on an interface, you cannot apply the **switchport voice vlan** command on the interface. If the **switchport voice vlan** *vlan-id* is already configured on an interface, you can apply a network-policy profile on the interface. This way the interface has the voice or voice-signaling VLAN network-policy profile applied on the interface.
- You cannot configure static secure MAC addresses on an interface that has a network-policy profile.
- When Cisco Discovery Protocol and LLDP are both in use within the same switch, it is necessary to disable LLDP on interfaces where Cisco Discovery Protocol is in use for power negotiation. LLDP can be disabled at interface level with the commands **no lldp tlv-select power-management** or **no lldp transmit / no lldp receive**.

Information About LLDP, LLDP-MED, and Wired Location Service

LLDP

The Cisco Discovery Protocol (CDP) is a device discovery protocol that runs over Layer 2 (the data link layer) on all Cisco-manufactured devices (routers, bridges, access servers, switches, and controllers). CDP allows

network management applications to automatically discover and learn about other Cisco devices connected to the network.

To support non-Cisco devices and to allow for interoperability between other devices, the device supports the IEEE 802.1AB Link Layer Discovery Protocol (LLDP). LLDP is a neighbor discovery protocol that is used for network devices to advertise information about themselves to other devices on the network. This protocol runs over the data-link layer, which allows two systems running different network layer protocols to learn about each other.

LLDP Supported TLVs

LLDP supports a set of attributes that it uses to discover neighbor devices. These attributes contain type, length, and value descriptions and are referred to as TLVs. LLDP supported devices can use TLVs to receive and send information to their neighbors. This protocol can advertise details such as configuration information, device capabilities, and device identity.

The switch supports these basic management TLVs. These are mandatory LLDP TLVs.

- Port description TLV
- System name TLV
- System description TLV
- System capabilities TLV
- Management address TLV

These organizationally specific LLDP TLVs are also advertised to support LLDP-MED.

- Port VLAN ID TLV (IEEE 802.1 organizationally specific TLVs)
- MAC/PHY configuration/status TLV (IEEE 802.3 organizationally specific TLVs)

LLDP and Cisco Device Stacks

A device stack appears as a single device in the network. Therefore, LLDP discovers the device stack, not the individual stack members.

LLDP-MED

LLDP for Media Endpoint Devices (LLDP-MED) is an extension to LLDP that operates between endpoint devices such as IP phones and network devices. It specifically provides support for voice over IP (VoIP) applications and provides additional TLVs for capabilities discovery, network policy, Power over Ethernet, inventory management and location information. By default, all LLDP-MED TLVs are enabled.

LLDP-MED Supported TLVs

LLDP-MED supports these TLVs:

- LLDP-MED capabilities TLV

  Allows LLDP-MED endpoints to determine the capabilities that the connected device supports and has enabled.

- Network policy TLV

Allows both network connectivity devices and endpoints to advertise VLAN configurations and associated Layer 2 and Layer 3 attributes for the specific application on that port. For example, the switch can notify a phone of the VLAN number that it should use. The phone can connect to any device, obtain its VLAN number, and then start communicating with the call control.

By defining a network-policy profile TLV, you can create a profile for voice and voice-signaling by specifying the values for VLAN, class of service (CoS), differentiated services code point (DSCP), and tagging mode. These profile attributes are then maintained centrally on the switch and propagated to the phone.

- Power management TLV

  Enables advanced power management between LLDP-MED endpoint and network connectivity devices. Allows devices and phones to convey power information, such as how the device is powered, power priority, and how much power the device needs.

  LLDP-MED also supports an extended power TLV to advertise fine-grained power requirements, end-point power priority, and end-point and network connectivity-device power status. LLDP is enabled and power is applied to a port, the power TLV determines the actual power requirement of the endpoint device so that the system power budget can be adjusted accordingly. The device processes the requests and either grants or denies power based on the current power budget. If the request is granted, the switch updates the power budget. If the request is denied, the device turns off power to the port, generates a syslog message, and updates the power budget. If LLDP-MED is disabled or if the endpoint does not support the LLDP-MED power TLV, the initial allocation value is used throughout the duration of the connection.

  You can change power settings by entering the **power inline** {**auto** [**max** *max-wattage*] | **never** | **static** [**max** *max-wattage*]} interface configuration command. By default the PoE interface is in **auto** mode; If no value is specified, the maximum is allowed (30 W).

- Inventory management TLV

  Allows an endpoint to send detailed inventory information about itself to the device, including information hardware revision, firmware version, software version, serial number, manufacturer name, model name, and asset ID TLV.

- Location TLV

  Provides location information from the device to the endpoint device. The location TLV can send this information:

  - Civic location information

    Provides the civic address information and postal information. Examples of civic location information are street address, road name, and postal community name information.

  - ELIN location information

    Provides the location information of a caller. The location is determined by the Emergency location identifier number (ELIN), which is a phone number that routes an emergency call to the local public safety answering point (PSAP) and which the PSAP can use to call back the emergency caller.

  - Geographic location information

    Provides the geographical details of a switch location such as latitude, longitude, and altitude of a switch.

  - custom location

    Provides customized name and value of a switch location.

Wired Location Service

The device uses the location service feature to send location and attachment tracking information for its connected devices to a Cisco Mobility Services Engine (MSE). The tracked device can be a wireless endpoint, a wired endpoint, or a wired device or controller. The device notifies the MSE of device link up and link down events through the Network Mobility Services Protocol (NMSP) location and attachment notifications.

The MSE starts the NMSP connection to the device, which opens a server port. When the MSE connects to the device there are a set of message exchanges to establish version compatibility and service exchange information followed by location information synchronization. After connection, the device periodically sends location and attachment notifications to the MSE. Any link up or link down events detected during an interval are aggregated and sent at the end of the interval.

When the device determines the presence or absence of a device on a link-up or link-down event, it obtains the client-specific information such as the MAC address, IP address, and username. If the client is LLDP-MED- or CDP-capable, the device obtains the serial number and UDI through the LLDP-MED location TLV or CDP.

Depending on the device capabilities, the device obtains this client information at link up:

- Slot and port specified in port connection
- MAC address specified in the client MAC address
- IP address specified in port connection
- 802.1X username if applicable
- Device category is specified as a *wired station*
- State is specified as *new*
- Serial number, UDI
- Model number
- Time in seconds since the device detected the association

Depending on the device capabilities, the device obtains this client information at link down:

- Slot and port that was disconnected
- MAC address
- IP address
- 802.1X username if applicable
- Device category is specified as a *wired station*
- State is specified as *delete*
- Serial number, UDI
- Time in seconds since the device detected the disassociation

When the device shuts down, it sends an attachment notification with the state *delete* and the IP address before closing the NMSP connection to the MSE. The MSE interprets this notification as disassociation for all the wired clients associated with the device.

If you change a location address on the device, the device sends an NMSP location notification message that identifies the affected ports and the changed address information.

Default LLDP Configuration

Table 5: Default LLDP Configuration

Feature	Default Setting
LLDP global state	Disabled
LLDP holdtime (before discarding)	120 seconds
LLDP timer (packet update frequency)	30 seconds
LLDP reinitialization delay	2 seconds
LLDP tlv-select	Disabled to send and receive all TLVs
LLDP interface state	Disabled
LLDP receive	Disabled
LLDP transmit	Disabled
LLDP med-tlv-select	Disabled to send all LLDP-MED TLVs. When LLDP is globally enabled, LLDP-MED-TLV is also enabled.

How to Configure LLDP, LLDP-MED, and Wired Location Service

Enabling LLDP

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**lldp run** **Example:** `Device (config)# lldp run`	Enables LLDP globally on the device.
Step 4	**interface** *interface-id* **Example:** `Device (config)# interface gigabitethernet 2/0/1`	Specifies the interface on which you are enabling LLDP, and enter interface configuration mode.
Step 5	**lldp transmit** **Example:** `Device(config-if)# lldp transmit`	Enables the interface to send LLDP packets.
Step 6	**lldp receive** **Example:** `Device(config-if)# lldp receive`	Enables the interface to receive LLDP packets.
Step 7	**end** **Example:** `Device(config-if)# end`	Returns to privileged EXEC mode.
Step 8	**show lldp** **Example:** `Device# show lldp`	Verifies the configuration.
Step 9	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring LLDP Characteristics

You can configure the frequency of LLDP updates, the amount of time to hold the information before discarding it, and the initialization delay time. You can also select the LLDP and LLDP-MED TLVs to send and receive.

Note Steps 3 through 6 are optional and can be performed in any order.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**lldp holdtime** *seconds* **Example:** `Device(config)# lldp holdtime 120`	(Optional) Specifies the amount of time a receiving device should hold the information from your device before discarding it. The range is 0 to 65535 seconds; the default is 120 seconds.
Step 4	**lldp reinit** *delay* **Example:** `Device(config)# lldp reinit 2`	(Optional) Specifies the delay time in seconds for LLDP to initialize on an interface. The range is 2 to 5 seconds; the default is 2 seconds.
Step 5	**lldp timer** *rate* **Example:** `Device(config)# lldp timer 30`	(Optional) Sets the sending frequency of LLDP updates in seconds. The range is 5 to 65534 seconds; the default is 30 seconds.
Step 6	**lldp tlv-select** **Example:** `Device(config)# tlv-select`	(Optional) Specifies the LLDP TLVs to send or receive.
Step 7	**interface** *interface-id* **Example:** `Device (config)# interface gigabitethernet 2/0/1`	Specifies the interface on which you are enabling LLDP, and enter interface configuration mode.

	Command or Action	Purpose
Step 8	**lldp med-tlv-select** **Example:** `Device (config-if)# lldp med-tlv-select inventory management`	(Optional) Specifies the LLDP-MED TLVs to send or receive.
Step 9	**end** **Example:** `Device (config-if)# end`	Returns to privileged EXEC mode.
Step 10	**show lldp** **Example:** `Device# show lldp`	Verifies the configuration.
Step 11	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring LLDP-MED TLVs

By default, the device only sends LLDP packets until it receives LLDP-MED packets from the end device. It then sends LLDP packets with MED TLVs, as well. When the LLDP-MED entry has been aged out, it again only sends LLDP packets.

By using the **lldp** interface configuration command, you can configure the interface not to send the TLVs listed in the following table.

Table 6: LLDP-MED TLVs

LLDP-MED TLV	Description
inventory-management	LLDP-MED inventory management TLV
location	LLDP-MED location TLV
network-policy	LLDP-MED network policy TLV
power-management	LLDP-MED power management TLV

Follow these steps to enable a TLV on an interface:

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**interface** *interface-id* **Example:** `Device (config)# interface gigabitethernet 2/0/1`	Specifies the interface on which you are enabling LLDP, and enter interface configuration mode.
Step 4	**lldp med-tlv-select** **Example:** `Device(config-if)# lldp med-tlv-select inventory management`	Specifies the TLV to enable.
Step 5	**end** **Example:** `Device(config-if)# end`	Returns to privileged EXEC mode.
Step 6	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring Network-Policy TLV

Procedure

	Command or Action	Purpose
Step 1	**enable**	Enables privileged EXEC mode.

	Command or Action	Purpose
	Example: `Device> enable`	• Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**network-policy profile** *profile number* **Example:** `Device(config)# network-policy profile 1`	Specifies the network-policy profile number, and enter network-policy configuration mode. The range is 1 to 4294967295.
Step 4	{**voice** | **voice-signaling**} **vlan** [*vlan-id* {**cos** *cvalue* | **dscp** *dvalue*}] | [[**dot1p** {**cos** *cvalue* | **dscp** *dvalue*}] | **none** | **untagged**] **Example:** `Device(config-network-policy)# voice vlan 100 cos 4`	Configures the policy attributes: • **voice**—Specifies the voice application type. • **voice-signaling**—Specifies the voice-signaling application type. • **vlan**—Specifies the native VLAN for voice traffic. • *vlan-id*—(Optional) Specifies the VLAN for voice traffic. The range is 1 to 4094. • **cos** *cvalue*—(Optional) Specifies the Layer 2 priority class of service (CoS) for the configured VLAN. The range is 0 to 7; the default is 5. • **dscp** *dvalue*—(Optional) Specifies the differentiated services code point (DSCP) value for the configured VLAN. The range is 0 to 63; the default is 46. • **dot1p**—(Optional) Configures the telephone to use IEEE 802.1p priority tagging and use VLAN 0 (the native VLAN). • **none**—(Optional) Do not instruct the IP telephone about the voice VLAN. The telephone uses the configuration from the telephone key pad.

	Command or Action	Purpose
		• **untagged**—(Optional) Configures the telephone to send untagged voice traffic. This is the default for the telephone. • **untagged**—(Optional) Configures the telephone to send untagged voice traffic. This is the default for the telephone.
Step 5	**exit** **Example:** `Device(config)# exit`	Returns to global configuration mode.
Step 6	**interface** *interface-id* **Example:** `Device (config)# interface gigabitethernet 2/0/1`	Specifies the interface on which you are configuring a network-policy profile, and enter interface configuration mode.
Step 7	**network-policy** *profile number* **Example:** `Device(config-if)# network-policy 1`	Specifies the network-policy profile number.
Step 8	**lldp med-tlv-select network-policy** **Example:** `Device(config-if)# lldp med-tlv-select network-policy`	Specifies the network-policy TLV.
Step 9	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.
Step 10	**show network-policy profile** **Example:** `Device# show network-policy profile`	Verifies the configuration.
Step 11	**copy running-config startup-config** **Example:**	(Optional) Saves your entries in the configuration file.

	Command or Action	Purpose
	`Device# copy running-config startup-config`	

Configuring Location TLV and Wired Location Service

Beginning in privileged EXEC mode, follow these steps to configure location information for an endpoint and to apply it to an interface.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.

Step 2	**location** {**admin-tag** *string* | **civic-location identifier** {*id* | **host**} | **elin-location** *string* **identifier** *id* | **custom-location identifier** {*id* | **host**} | **geo-location identifier** {*id* | **host**}}
	Example:
	`Device(config)# location civic-location identifier 1`
	`Device(config-civic)# number 3550`
	`Device(config-civic)# primary-road-name "Cisco Way"`
	`Device(config-civic)# city "San Jose"`
	`Device(config-civic)# state CA`
	`Device(config-civic)# building 19`
	`Device(config-civic)# room C6`
	`Device(config-civic)# county "Santa Clara"`
	`Device(config-civic)# country US`
		Specifies the location information for an endpoint.
		• **admin-tag**—Specifies an administrative tag or site information.
		• **civic-location**—Specifies civic location information.
		• **elin-location**—Specifies emergency location information (ELIN).
		• **custom-location**—Specifies custom location information.
		• **geo-location**—Specifies geo-spatial location information.
		• **identifier** *id*—Specifies the ID for the civic, ELIN, custom, or geo location.
		• **host**—Specifies the host civic, custom, or geo location.
		• *string*—Specifies the site or location information in alphanumeric format.

Step 3	**exit** **Example:** `Device(config-civic)# exit`	Returns to global configuration mode.

	Command or Action	Purpose
Step 4	**interface** *interface-id* **Example:** `Device (config)# interface gigabitethernet2/0/1`	Specifies the interface on which you are configuring the location information, and enter interface configuration mode.
Step 5	**location** {**additional-location-information** *word* | **civic-location-id** {*id* | **host**} | **elin-location-id** *id* | **custom-location-id** {*id* | **host**} | **geo-location-id** {*id* | **host**} } **Example:** `Device(config-if)# location elin-location-id 1`	Enters location information for an interface: • **additional-location-information**—Specifies additional information for a location or place. • **civic-location-id**—Specifies global civic location information for an interface. • **elin-location-id**—Specifies emergency location information for an interface. • **custom-location-id**—Specifies custom location information for an interface. • **geo-location-id**—Specifies geo-spatial location information for an interface. • **host**—Specifies the host location identifier. • *word*—Specifies a word or phrase with additional location information. • *id*—Specifies the ID for the civic, ELIN, custom, or geo location. The ID range is 1 to 4095.
Step 6	**end** **Example:** `Device(config-if)# end`	Returns to privileged EXEC mode.
Step 7	Use one of the following: • **show location admin-tag** *string* • **show location civic-location identifier** *id* • **show location elin-location identifier** *id* **Example:** `Device# show location admin-tag` or	Verifies the configuration.

	Command or Action	Purpose

	`Device# show location civic-location identifier`
	
	or
	
	`Device# show location elin-location identifier`	

Step 8	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Enabling Wired Location Service on the Device

Before you begin

For wired location to function, you must first enter the **ip device tracking** global configuration command.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**nmsp notification interval** {**attachment** | **location**} *interval-seconds* **Example:** `Device(config)# nmsp notification interval location 10`	Specifies the NMSP notification interval. **attachment**—Specifies the attachment notification interval. **location**—Specifies the location notification interval. *interval-seconds*—Duration in seconds before the device sends the MSE the location or attachment updates. The range is 1 to 30; the default is 30.

	Command or Action	Purpose
Step 4	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.
Step 5	**show network-policy profile** **Example:** `Device# show network-policy profile`	Verifies the configuration.
Step 6	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuration Examples for LLDP, LLDP-MED, and Wired Location Service

Configuring Network-Policy TLV: Examples

This example shows how to configure VLAN 100 for voice application with CoS and to enable the network-policy profile and network-policy TLV on an interface:

```
# configure terminal
(config)# network-policy 1
(config-network-policy)# voice vlan 100 cos 4
(config-network-policy)# exit
(config)# interface gigabitethernet 1/0/1
(config-if)# network-policy profile 1
(config-if)# lldp med-tlv-select network-policy
```

This example shows how to configure the voice application type for the native VLAN with priority tagging:

```
config-network-policy)# voice vlan dot1p cos 4
config-network-policy)# voice vlan dot1p dscp 34
```

Monitoring and Maintaining LLDP, LLDP-MED, and Wired Location Service

Commands for monitoring and maintaining LLDP, LLDP-MED, and wired location service.

Command	Description
clear lldp counters	Resets the traffic counters to zero.
clear lldp table	Deletes the LLDP neighbor information table.
clear nmsp statistics	Clears the NMSP statistic counters.
show lldp	Displays global information, such as frequency of transmissions, the holdtime for packets being sent, and the delay time before LLDP initializes on an interface.
show lldp entry *entry-name*	Displays information about a specific neighbor. You can enter an asterisk (*) to display all neighbors, or you can enter the neighbor name.
show lldp interface [*interface-id*]	Displays information about interfaces with LLDP enabled. You can limit the display to a specific interface.
show lldp neighbors [*interface-id*] [**detail**]	Displays information about neighbors, including device type, interface type and number, holdtime settings, capabilities, and port ID. You can limit the display to neighbors of a specific interface or expand the display for more detailed information.
show lldp traffic	Displays LLDP counters, including the number of packets sent and received, number of packets discarded, and number of unrecognized TLVs.
show location admin-tag *string*	Displays the location information for the specified administrative tag or site.
show location civic-location identifier *id*	Displays the location information for a specific global civic location.
show location elin-location identifier *id*	Displays the location information for an emergency location
show network-policy profile	Displays the configured network-policy profiles.
show nmsp	Displays the NMSP information

Additional References for LLDP, LLDP-MED, and Wired Location Service

MIBs

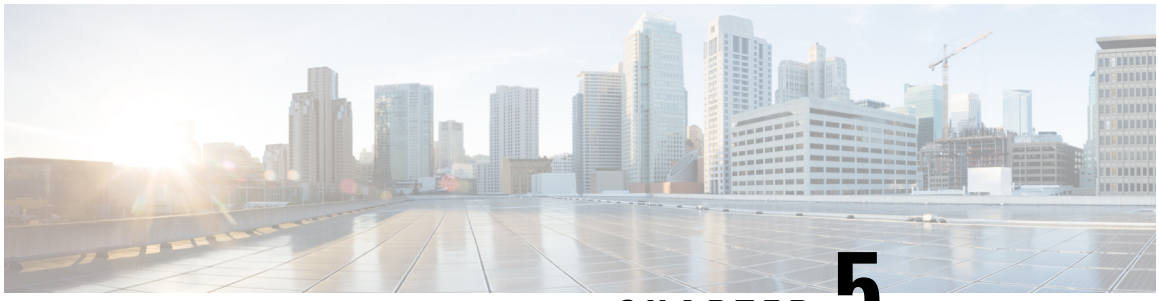
MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature Information for LLDP, LLDP-MED, and Wired Location Service

Release	Modification
Cisco IOS XE 3.2SE	This feature was introduced.



CHAPTER 5

Configuring System MTU

- Restrictions for System MTU, on page 61
- Information About the MTU, on page 61
- How to Configure MTU Sizes, on page 62
- Configuration Examples for System MTU, on page 64
- Additional References for System MTU, on page 64
- Feature Information for System MTU, on page 65

Restrictions for System MTU

When configuring the system MTU values, follow these guidelines:

- The device does not support the MTU on a per-interface basis.
- If you enter the **system mtu** *bytes* global configuration command, the command affects all the switched and routed ports on the switch.

Information About the MTU

The default maximum transmission unit (MTU) size for frames received and sent on all device interfaces is 1500 bytes.

System MTU Value Application

In a switch stack, the MTU values applied to member switches depends upon the stack configuration. The following stack configurations are supported:

This table shows how the MTU values are applied.

Table 7: MTU Values

Configuration	system mtu command	ip mtu command	ipv6 mtu command
Standalone switch or switch stack	You can enter the **system mtu** command on a switch or switch stack, but system MTU value does not take effect on the switch. It affects the Fast Ethernet ports. The range is from 1500 to 9198 bytes.	Use the **ip mtu** *bytes* command. The range is from 832 up to 1500 bytes. **Note** The IP MTU value is the applied value, not the configured value.	Use the **ipv6 mtu** *bytes* command. The range is from 1280 to the system jumbo MTU value (in bytes). **Note** The IPv6 MTU value is the applied value, not the configured value.

The upper limit of the IP or IPv6 MTU value is based on the switch or switch stack configuration and refers to the currently applied system MTU value. For more information about setting the MTU sizes, see the **system mtu** global configuration command in the command reference for this release.

How to Configure MTU Sizes

Configuring the System MTU

Follow these steps to change the MTU size for switched packets:

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**system mtu** *bytes* **Example:** `Device(config)# system mtu 1900`	(Optional) Changes the MTU size for all Gigabit Ethernet and 10-Gigabit Ethernet interfaces.

	Command or Action	Purpose
Step 4	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.
Step 5	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	Saves your entries in the configuration file.
Step 6	**show system mtu** **Example:** `Device# show system mtu`	Verifies your settings.

Configuring Protocol-Specific MTU

To override system MTU values on routed interfaces, configure protocol-specific MTU under each routed interface.

Beginning in privileged EXEC mode, follow these steps to change the MTU size for routed ports:

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface* **Example:** `Device(config)# interface gigabitethernet0/0`	Enters interface configuration mode.
Step 3	**ip mtu** *bytes* **Example:** `Device(config-if)# ip mtu 68`	Changes the IPv4 MTU size
Step 4	**ipv6 mtu** *bytes* **Example:** `Device(config-if)# ipv6 mtu 1280`	(Optional) Changes the IPv6 MTU size.
Step 5	**end** **Example:** `Device(config-if)# end`	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 6	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	Saves your entries in the configuration file.
Step 7	**show system mtu** **Example:** `Device# show system mtu`	Verifies your settings.

Configuration Examples for System MTU

Example: Configuring Protocol-Specific MTU

```
Device# configure terminal
Device(config)# interface gigabitethernet 0/0
Device(config-if)# ip mtu 900
Device(config-if)# ipv6 mtu 1286
Device(config-if)# end
```

Example: Configuring the System MTU

```
Device# configure terminal
Device(config)# system mtu 1600
Device(config)# exit
```

Additional References for System MTU

MIBs

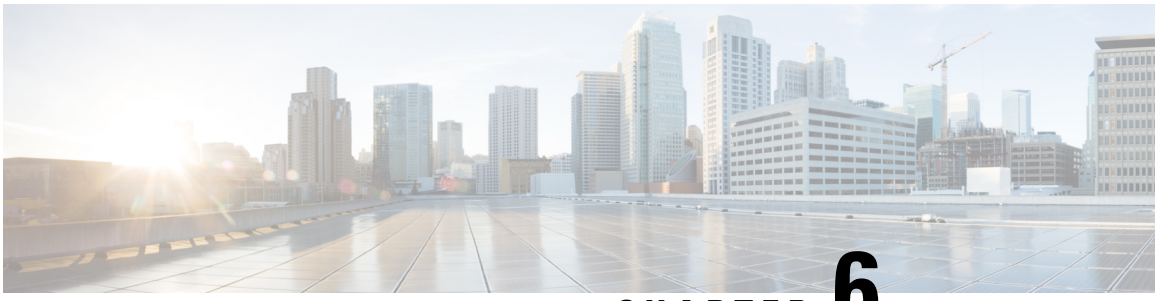
MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature Information for System MTU

Release	Modification
Cisco IOS XE 3.2SE	This feature was introduced.



CHAPTER 6

Configuring Internal Power Supplies

- Information About Internal Power Supplies , on page 67
- How to Configure Internal Power Supplies, on page 67
- Monitoring Internal Power Supplies, on page 68
- Configuration Examples for Internal Power Supplies, on page 68
- Additional References for Internal Power Supplies, on page 69
- Feature History and Information for Internal Power Supplies, on page 70

Information About Internal Power Supplies

See the device installation guide for information about the power supplies.

How to Configure Internal Power Supplies

Configuring Internal Power Supply

You can use the **power supply** EXEC command to configure and manage the internal power supply on the device. The device does not support the **no power supply** EXEC command.

Follow these steps beginning in user EXEC mode:

Procedure

	Command or Action	Purpose
Step 1	**power supply** *switch_number* **slot**{**A** | **B**} { **off** | **on** } **Example:** `Device# power supply 1 slot A on`	Sets the specified power supply to **off** or **on** by using one of these keywords: • **A** —Selects the power supply in slot A. • **B** —Selects power supply in slot B. **Note** Power supply slot B is the closest to the outer edge of the device.

	Command or Action	Purpose
		• **off** —Set the power supply off. • **on** —Set the power supply on. By default, the device power supply is **on**.
Step 2	**show environment power** **Example:** `Device# show environment power`	Verifies your settings.

Monitoring Internal Power Supplies

Table 8: Show Commands for Power Supplies

Command	Purpose
show environment power [**all** | **switch** *switch_number*]	(Optional) Displays the status of the internal power supplies for each device in the stack or for the specified device. The range is 1 to 9, depending on the device member numbers in the stack. The device keywords are available only on stacking-capable devices.

Configuration Examples for Internal Power Supplies

This example shows how to set the power supply in slot A to off:

```
Device# power supply 1 slot A off
Disabling Power supply A may result in a power loss to PoE devices and/or switches ...
Continue? (yes/[no]): yes
Device#
Jun 10 04:52:54.389: %PLATFORM_ENV-6-FRU_PS_OIR: FRU Power Supply 1 powered off
Jun 10 04:52:56.717: %PLATFORM_ENV-1-FAN_NOT_PRESENT: Fan is not present
Device#
```

This example shows how to set the power supply in slot A to on:

```
Device# power supply 1 slot A on
Jun 10 04:54:39.600: %PLATFORM_ENV-6-FRU_PS_OIR: FRU Power Supply 1 powered on
```

This example shows the output of the **show env power** command:

```
Device# show env power

SW  PID                 Serial#     Status           Sys Pwr  PoE Pwr  Watts
--- ------------------  ----------  ---------------  -------  -------  -----
1A  PWR-C1-715WAC       LIT161010UE  OK              Good     Good     715
```

```
1B  Not Present

Device#
```

Table 9: show env power Status Descriptions

Field	Description
OK	The power supply is present and power is good.
Not Present	No power supply is installed.
No Input Power	The power supply is present but there is no input power.
Disabled	The power supply and input power are present, but power supply is switched off by CLI.
Not Responding	The power supply is not recognizable or is faulty.
Failure-Fan	The power supply fan is faulty.

Additional References for Internal Power Supplies

MIBs

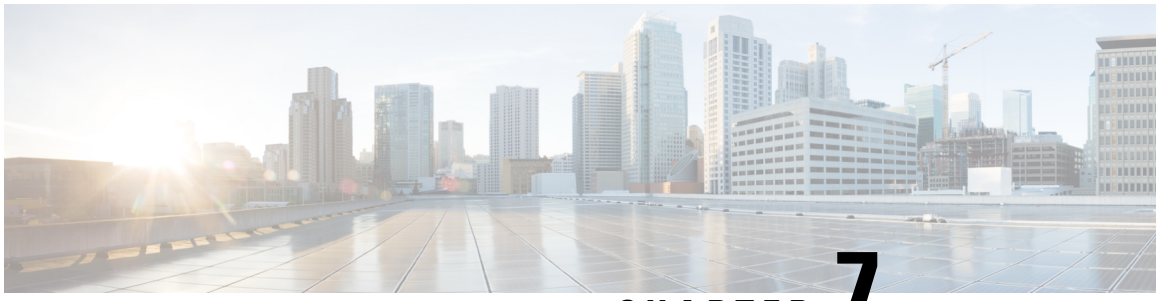
MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information for Internal Power Supplies

Release	Modification
Cisco IOS XE 3.2SE	This feature was introduced.
Cisco IOS XE 3.3SE	The **slot** keyword replaced the **frufep** keyword.



CHAPTER 7

Configuring Power over Ethernet

- Information About PoE, on page 71
- How to Configure PoE and UPoE, on page 76
- Monitoring Power Status, on page 81
- Additional References, on page 81
- Feature Information for PoE, on page 82

Information About PoE

PoE and PoE+ Ports

A PoE-capable switch port automatically supplies power to one of these connected devices if the device senses that there is no power on the circuit:

- A Cisco prestandard powered device (such as a Cisco IP Phone)
- An IEEE 802.3af-compliant powered device
- An IEEE 802.3at-compliant powered device

A powered device can receive redundant power when it is connected to a PoE switch port and to an AC power source. The device does not receive redundant power when it is only connected to the PoE port.

Supported Protocols and Standards

The device uses these protocols and standards to support PoE:

- CDP with power consumption—The powered device notifies the device of the amount of power it is consuming. The device does not reply to the power-consumption messages. The device can only supply power to or remove power from the PoE port.
- Cisco intelligent power management—The powered device and the device negotiate through power-negotiation CDP messages for an agreed-upon power-consumption level. The negotiation allows a high-power Cisco powered device, which consumes more than 7 W, to operate at its highest power mode. The powered device first boots up in low-power mode, consumes less than 7 W, and negotiates to obtain enough power to operate in high-power mode. The device changes to high-power mode only when it receives confirmation from the device.

High-power devices can operate in low-power mode on devices that do not support power-negotiation CDP.

Cisco intelligent power management is backward-compatible with CDP with power consumption; the device responds according to the CDP message that it receives. CDP is not supported on third-party powered devices; therefore, the device uses the IEEE classification to determine the power usage of the device.

- IEEE 802.3af—The major features of this standard are powered-device discovery, power administration, disconnect detection, and optional powered-device power classification. For more information, see the standard.
- IEEE 802.3at—The PoE+ standard increases the maximum power that can be drawn by a powered device from 15.4 W per port to 30 W per port.
- The Cisco UPOE feature provides the capability to source up to 60 W of power (2 x 30 W) over both signal and spare pairs of the RJ-45 Ethernet cable by using the Layer-2 power negotiation protocols such as CDP or LLDP. An LLDP and CDP request of 30 W and higher in presence of the 4-wire Cisco Proprietary spare-pair power TLV can provide power on the spare pair.

Powered-Device Detection and Initial Power Allocation

The device detects a Cisco pre-standard or an IEEE-compliant powered device when the PoE-capable port is in the no-shutdown state, PoE is enabled (the default), and the connected device is not being powered by an AC adaptor.

After device detection, the device determines the device power requirements based on its type:

- The initial power allocation is the maximum amount of power that a powered device requires. The device initially allocates this amount of power when it detects and powers the powered device. As the device receives CDP messages from the powered device and as the powered device negotiates power levels with the device through CDP power-negotiation messages, the initial power allocation might be adjusted.
- The device classifies the detected IEEE device within a power consumption class. Based on the available power in the power budget, the device determines if a port can be powered. Table 10: IEEE Power Classifications, on page 72 lists these levels.

Table 10: IEEE Power Classifications

Class	Maximum Power Level Required from the Device
0 (class status unknown)	15.4 W
1	4 W
2	7 W
3	15.4 W
4	30 W (For IEEE 802.3at Type 2 powered devices)

The device monitors and tracks requests for power and grants power only when it is available. The device tracks its power budget (the amount of power available on the device for PoE). The device performs power-accounting calculations when a port is granted or denied power to keep the power budget up to date.

After power is applied to the port, the device uses CDP to determine the *CDP-specific* power consumption requirement of the connected Cisco powered devices, which is the amount of power to allocate based on the CDP messages. The device adjusts the power budget accordingly. This does not apply to third-party PoE devices. The device processes a request and either grants or denies power. If the request is granted, the device updates the power budget. If the request is denied, the device ensures that power to the port is turned off, generates a syslog message, and updates the LEDs. Powered devices can also negotiate with the device for more power.

With PoE+, powered devices use IEEE 802.3at and LLDP power with media dependent interface (MDI) type, length, and value descriptions (TLVs), Power-via-MDI TLVs, for negotiating power up to 30 W. Cisco pre-standard devices and Cisco IEEE powered devices can use CDP or the IEEE 802.3at power-via-MDI power negotiation mechanism to request power levels up to 30 W.

Note The initial allocation for Class 0, Class 3, and Class 4 powered devices is 15.4 W. When a device starts up and uses CDP or LLDP to send a request for more than 15.4 W, it can be allocated up to the maximum of 30 W.

Note The CDP-specific power consumption requirement is referred to as the *actual* power consumption requirement in the software configuration guides and command references.

If the device detects a fault caused by an undervoltage, overvoltage, overtemperature, oscillator-fault, or short-circuit condition, it turns off power to the port, generates a syslog message, and updates the power budget and LEDs.

The PoE feature operates the same whether or not the device is a stack member. The power budget is per device and independent of any other device in the stack. Election of a new active device does not affect PoE operation. The active device keeps track of the PoE status for all devices and ports in the stack and includes the status in output displays.

The stacking-capable device also supports StackPower, which allows the power supplies to share the load across multiple systems in a stack when you connect the devices with power stack cables. You can manage the power supplies of up to four stack members as a one large power supply.

Power Management Modes

The device supports these PoE modes:

- **auto**—The device automatically detects if the connected device requires power. If the device discovers a powered device connected to the port and if the device has enough power, it grants power, updates the power budget, turns on power to the port on a first-come, first-served basis, and updates the LEDs. For LED information, see the hardware installation guide.

  If the device has enough power for all the powered devices, they all come up. If enough power is available for all powered devices connected to the device, power is turned on to all devices. If there is not enough available PoE, or if a device is disconnected and reconnected while other devices are waiting for power, it cannot be determined which devices are granted or are denied power.

  If granting power would exceed the system power budget, the device denies power, ensures that power to the port is turned off, generates a syslog message, and updates the LEDs. After power has been denied, the device periodically rechecks the power budget and continues to attempt to grant the request for power.

If a device being powered by the device is then connected to wall power, the device might continue to power the device. The device might continue to report that it is still powering the device whether the device is being powered by the device or receiving power from an AC power source.

If a powered device is removed, the device automatically detects the disconnect and removes power from the port. You can connect a nonpowered device without damaging it.

You can specify the maximum wattage that is allowed on the port. If the IEEE class maximum wattage of the powered device is greater than the configured maximum value, the device does not provide power to the port. If the device powers a powered device, but the powered device later requests through CDP messages more than the configured maximum value, the device removes power to the port. The power that was allocated to the powered device is reclaimed into the global power budget. If you do not specify a wattage, the device delivers the maximum value. Use the **auto** setting on any PoE port. The auto mode is the default setting.

- **static**—The device pre-allocates power to the port (even when no powered device is connected) and guarantees that power will be available for the port. The device allocates the port configured maximum wattage, and the amount is never adjusted through the IEEE class or by CDP messages from the powered device. Because power is pre-allocated, any powered device that uses less than or equal to the maximum wattage is guaranteed to be powered when it is connected to the static port. The port no longer participates in the first-come, first-served model.

  However, if the powered-device IEEE class is greater than the maximum wattage, the device does not supply power to it. If the device learns through CDP messages that the powered device is consuming more than the maximum wattage, the device shuts down the powered device.

  If you do not specify a wattage, the device pre-allocates the maximum value. The device powers the port only if it discovers a powered device. Use the **static** setting on a high-priority interface.

- **never**—The device disables powered-device detection and never powers the PoE port even if an unpowered device is connected. Use this mode only when you want to make sure that power is never applied to a PoE-capable port, making the port a data-only port.

For most situations, the default configuration (auto mode) works well, providing plug-and-play operation. No further configuration is required. However, perform this task to configure a PoE port for a higher priority, to make it data only, or to specify a maximum wattage to disallow high-power powered devices on a port.

Stacking-capable devices also support StackPower, which allows device power supplies to share the load across multiple systems in a stack by connecting up to four devices with power stack cables.

Power Monitoring and Power Policing

When policing of the real-time power consumption is enabled, the device takes action when a powered device consumes more power than the maximum amount allocated, also referred to as the *cutoff-power value*.

When PoE is enabled, the device senses the real-time power consumption of the powered device. The device monitors the real-time power consumption of the connected powered device; this is called *power monitoring* or *power sensing*. The device also polices the power usage with the *power policing* feature.

Power monitoring is backward-compatible with Cisco intelligent power management and CDP-based power consumption. It works with these features to ensure that the PoE port can supply power to the powered device.

The device senses the real-time power consumption of the connected device as follows:

1. The device monitors the real-time power consumption on individual ports.

2. The device records the power consumption, including peak power usage. The device reports the information through the CISCO-POWER-ETHERNET-EXT-MIB.

3. If power policing is enabled, the device polices power usage by comparing the real-time power consumption to the maximum power allocated to the device. The maximum power consumption is also referred to as the *cutoff power* on a PoE port.

   If the device uses more than the maximum power allocation on the port, the device can either turn off power to the port, or the device can generate a syslog message and update the LEDs (the port LED is now blinking amber) while still providing power to the device based on the device configuration. By default, power-usage policing is disabled on all PoE ports.

   If error recovery from the PoE error-disabled state is enabled, the device automatically takes the PoE port out of the error-disabled state after the specified amount of time.

   If error recovery is disabled, you can manually re-enable the PoE port by using the **shutdown** and **no shutdown** interface configuration commands.

4. If policing is disabled, no action occurs when the powered device consumes more than the maximum power allocation on the PoE port, which could adversely affect the device.

Power Consumption Values

You can configure the initial power allocation and the maximum power allocation on a port. However, these values are only the configured values that determine when the device should turn on or turn off power on the PoE port. The maximum power allocation is not the same as the actual power consumption of the powered device. The actual cutoff power value that the device uses for power policing is not equal to the configured power value.

When power policing is enabled, the device polices the power usage *at the switch port*, which is greater than the power consumption of the device. When you manually set the maximum power allocation, you must consider the power loss over the cable from the switch port to the powered device. The cutoff power is the sum of the rated power consumption of the powered device and the worst-case power loss over the cable.

We recommend that you enable power policing when PoE is enabled on your device. For example, if policing is disabled and you set the cutoff-power value by using the **power inline auto max 6300** interface configuration command, the configured maximum power allocation on the PoE port is 6.3 W (6300 mW). The device provides power to the connected devices on the port if the device needs up to 6.3 W. If the CDP-power negotiated value or the IEEE classification value exceeds the configured cutoff value, the device does not provide power to the connected device. After the device turns on power on the PoE port, the device does not police the real-time power consumption of the device, and the device can consume more power than the maximum allocated amount, which could adversely affect the device and the devices connected to the other PoE ports.

Because a standalone device supports internal power supplies, the total amount of power available for the powered devices varies depending on the power supply configuration.

- If a power supply is removed and replaced by a new power supply with less power and the device does not have enough power for the powered devices, the device denies power to the PoE ports in auto mode in descending order of the port numbers. If the device still does not have enough power, the device then denies power to the PoE ports in static mode in descending order of the port numbers.

- If the new power supply supports more power than the previous one and the device now has more power available, the device grants power to the PoE ports in static mode in ascending order of the port numbers. If it still has power available, the device then grants power to the PoE ports in auto mode in ascending order of the port numbers.

The stacking-capable device also supports StackPower, which allows power supplies to share the load across multiple systems in a stack by connecting the devices with power stack cables. You can collectively manage the power supplies of up to four stack members as one large power supply.

Cisco Universal Power Over Ethernet

Cisco Universal Power Over Ethernet (Cisco UPOE) is a Cisco proprietary technology that extends the IEEE 802.at PoE standard to provide the capability to source up to 60 W of power over standard Ethernet cabling infrastructure (Class D or better) by using the spare pair of an RJ-45 cable (wires 4,5,7,8) with the signal pair (wires 1,2,3,6). Power on the spare pair is enabled when the switch port and end device mutually identify themselves as Cisco UPOE-capable using CDP or LLDP and the end device requests for power to be enabled on the spare pair. When the spare pair is powered, the end device can negotiate up to 60 W of power from the switch using CDP or LLDP.

If the end device is PoE-capable on both signal and spare pairs but does not support the CDP or LLDP extensions required for Cisco UPOE, a 4-pair forced mode configuration automatically enables power on both signal and spare pairs from the switch port.

How to Configure PoE and UPoE

Configuring a Power Management Mode on a PoE Port

Note When you make PoE configuration changes, the port being configured drops power. Depending on the new configuration, the state of the other PoE ports, and the state of the power budget, the port might not be powered up again. For example, port 1 is in the auto and on state, and you configure it for static mode. The device removes power from port 1, detects the powered device, and repowers the port. If port 1 is in the auto and on state and you configure it with a maximum wattage of 10 W, the device removes power from the port and then redetects the powered device. The device repowers the port only if the powered device is a class 1, class 2, or a Cisco-only powered device.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**interface** *interface-id* **Example:** `Device(config)# interface gigabitethernet 2/0/1`	Specifies the physical port to be configured, and enters interface configuration mode.
Step 4	**power inline** {**auto** [**max** *max-wattage*] | **never** | **static** [**max** *max-wattage*]} **Example:** `Device(config-if)# power inline auto`	Configures the PoE mode on the port. The keywords have these meanings: • **auto**—Enables powered-device detection. If enough power is available, automatically allocates power to the PoE port after device detection. This is the default setting. • **max** *max-wattage*—Limits the power allowed on the port. The range for Cisco UPOE ports is 4000 to 60000 mW. If no value is specified, the maximum is allowed. • **never** —Disables device detection, and disable power to the port. **Note** If a port has a Cisco powered device connected to it, do not use the **power inline never** command to configure the port. A false link-up can occur, placing the port into the error-disabled state. • **static**—Enables powered-device detection. Pre-allocate (reserve) power for a port before the device discovers the powered device. The device reserves power for this port even when no device is connected and guarantees that power will be provided upon device detection. The device allocates power to a port configured in static mode before it allocates power to a port configured in auto mode.
Step 5	**end** **Example:** `Device(config-if)# end`	Returns to privileged EXEC mode.
Step 6	**show power inline** [*interface-id* | **module** *switch-number*] **Example:** `Device# show power inline`	Displays PoE status for a device or a device stack, for the specified interface, or for a specified stack member.. The **module** *switch-number* keywords are supported only on stacking-capable devices.

	Command or Action	Purpose
Step 7	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Enabling Power on Signal/Spare Pairs

Note Do not enter this command if the end device cannot source inline power on the spare pair or if the end device supports the CDP or LLDP extensions for Cisco UPOE.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Device(config)# interface gigabitethernet2/0/1`	Specifies the physical port to be configured, and enters interface configuration mode.
Step 3	**power inline four-pair forced** **Example:** `Device(config-if)# power inline four-pair forced`	Enables power on both signal and spare pairs from a switch port.
Step 4	**end** **Example:** `Device(config-if)# end`	Returns to privileged EXEC mode.

Configuring Power Policing

By default, the device monitors the real-time power consumption of connected powered devices. You can configure the device to police the power usage. By default, policing is disabled.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**interface** *interface-id* **Example:** `Device(config)# interface gigabitethernet 2/0/1`	Specifies the physical port to be configured, and enter interface configuration mode.
Step 4	**power inline police** [**action** {**log** | **errdisable**}] **Example:** `Device(config-if)# power inline police`	If the real-time power consumption exceeds the maximum power allocation on the port, configures the device to take one of these actions: • **power inline police**—Shuts down the PoE port, turns off power to it, and puts it in the error-disabled state. **Note** You can enable error detection for the PoE error-disabled cause by using the **errdisable detect cause inline-power** global configuration command. You can also enable the timer to recover from the PoE error-disabled state by using the **errdisable recovery cause inline-power interval** *interval* global configuration command. • **power inline police action errdisable**—Turns off power to the port if the real-time power consumption exceeds the maximum power allocation on the port. • **power inline police action log**—Generates a syslog message while still providing power to the port. If you do not enter the **action log** keywords, the default action shuts down the port and puts the port in the error-disabled state.

	Command or Action	Purpose
Step 5	**exit** **Example:** `Device(config-if)# exit`	Returns to global configuration mode.

Step 6	Use one of the following:
	• **errdisable detect cause inline-power**
	• **errdisable recovery cause inline-power**
	• **errdisable recovery interval** *interval*
	Example:
	`Device(config)# errdisable detect cause inline-power`
	`Device(config)# errdisable recovery cause inline-power`
	`Device(config)# errdisable recovery interval 100`
		(Optional) Enables error recovery from the PoE error-disabled state, and configures the PoE recover mechanism variables.
		By default, the recovery interval is 300 seconds.
		For **interval** *interval*, specifies the time in seconds to recover from the error-disabled state. The range is 30 to 86400.

Step 7	**exit** **Example:** `Device(config)# exit`	Returns to privileged EXEC mode.

Step 8	Use one of the following:
	• **show power inline police**
	• **show errdisable recovery**
	Example:
	`Device# show power inline police`
	`Device# show errdisable recovery`
		Displays the power monitoring status, and verify the error recovery settings.

Step 9	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Monitoring Power Status

Table 11: Show Commands for Power Status

Command	Purpose
show env power switch [*switch-number*]	(Optional) Displays the status of the internal power supplies for each switch in the stack or for the specified switch. The range is 1 to 9, depending on the switch member numbers in the stack. These keywords are available only on stacking-capable switches.
show power inline [*interface-id* | **module** *switch-number*]	Displays PoE status for a switch or switch stack, for an interface, or for a specific switch in the stack.
show power inline police	Displays the power policing data.

Additional References

MIBs

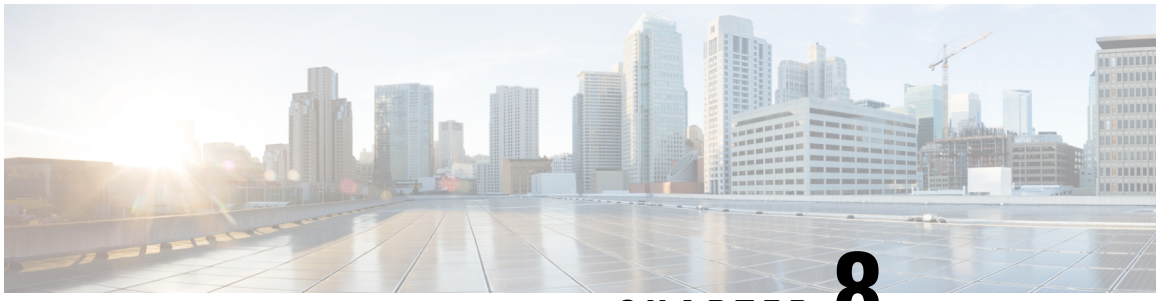
MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature Information for PoE

Release	Modification
Cisco IOS XE 3.2SE	This feature was introduced.
Cisco IOS XE 3.3SE	The **four-pair forced** keywords were added.



CHAPTER 8

Configuring the Cisco Expandable Power System 2200

This module contains the following sections:

- Restrictions for Configuring the XPS 2200, on page 83
- Information About Configuring the XPS 2200, on page 83
- How to Configure the XPS 2200, on page 87
- Monitoring and Maintaining the XPS 2200, on page 90
- Additional References for XPS 2200, on page 90
- Feature History and Information for XPS 2200, on page 91

Restrictions for Configuring the XPS 2200

- When using the XPS power supplies in the RPS mode for backing up switch power supplies, the smallest power supply in the XPS must be greater than the largest power supply in a switch connected to an XPS port in RPS mode.
- In RPS mode, each XPS power supply can back up one and only one switch power supply, regardless of the size.
- If you remove a power supply from the power stack (from a switch or the XPS), be sure that removing it does not deplete available power enough to cause load shedding.

Information About Configuring the XPS 2200

Cisco eXpandable Power System (XPS) 2200 Overview

The Cisco eXpandable Power System (XPS) 2200 is a standalone power system that you can connect to Catalyst switches. The XPS 2200 can provide backup power to connected devices that experience a power supply failure or, in a Catalyst switch power stack, it can supply additional power to the power stack budget. The XPS 2200 power ports and internal power supplies can operate in redundant power supply (RPS) mode or stack power (SP) mode.

Stack-power mode is used only on stacking-capable switches in a power stack. With no XPS, a power stack operates in ring topology with a maximum of four switches in the stack. If you merge two stacks, the total number of switches cannot exceed four. When an XPS is in the power stack, you can connect up to nine switches in the stack plus the XPS, providing power budgets to power stack members similar to stack-power ring topology operation.

All Catalyst switches connected to an XPS on SP ports are part of the same power stack, and all power from the XPS and the switch is shared across all switches in the stack. Power sharing is the default mode, but the XPS supports the same stack power modes that are supported in a ring topology (strict and nonstrict power-sharing or redundant modes).

When two power supplies are present, the system can operate in mixed mode, where one power supply operates in RPS mode and the other in SP mode. You can configure the ports and power supplies for the way that you plan to use the XPS 2200.

The XPS 2200 has nine power ports that can operate in an RPS role or in an automatic stack power (Auto-SP) role (the default), where mode of operation is determined by the type of switch connected to the port . You can also use the CLI to force the mode to be RPS for stackable switches.

-
- When a Catalyst switch running the LAN Base image, or a Catalyst (nonstackable) switch is connected to the port, the mode is RPS, and the XPS 2200 acts as a backup if the switch power supply goes down.
- When a Catalyst (stackable) switch running the IP Base image or IP Services image is connected to the port, the mode is SP, which enables the switch to be part of the stack power system.

You configure the XPS through any switch connected to a power port. You can use any XPS port for configuration, and you can configure any port from any switch connected to the XPS. If you enter XPS configuration commands on more than one switch, the last configuration applied takes effect.

Although all XPS configuration is done through a switch, the XPS 2200 also runs its own software. You can upgrade this software through the XPS Service Port.

XPS 2200 Power Supply Modes

The XPS has two power supplies that can also be in either RPS or SP mode.

In SP mode, all SP ports on the XPS belong to the same power stack. When a power stack includes an XPS, the stack topology is a star topology and consists of up to nine member switches plus the XPS 2200. The XPS power supply or power supplies that are in SP mode are considered in the power budgeting. If both XPS power supplies are in RPS mode, the power stack consists only of the switches connected to XPS ports in SP mode, and the power budget is determined by the power supplies in these switches.

If there is a power supply role mismatch, for example, if an XPS port is configured for RPS and both power supplies are in SP mode, the XPS detects the mismatch, and an error message is sent.

RPS Mode

When both XPS power supplies are in RPS mode, the XPS can back up two power supply failures for switch power supplies of equal value or less. The smallest power supply in the XPS must be greater than the largest power supply in a switch connected to an XPS port in RPS mode.

If only one supply is in RPS mode, the XPS can back up only one power supply, even when the failed power supply is much smaller. For example, if an XPS 1100 W power supply is in RPS mode and two 350 W switch power supplies fail, the XPS can back up only one of the switch power supplies.

When one XPS power supply in RPS mode is backing up a switch power supply and another switch power supply fails, a message appears that the XPS backup is not available. When the failed power supply comes up, the XPS becomes available to back up other power supplies.

If the XPS is backing up two failed power supplies in a single switch (both XPS power supplies in RPS mode), the XPS is not available to back up other switch power supplies until both of the failed supplies are repaired or replaced.

In mixed mode, with one power supply in RPS mode and one in SP mode, if two power supplies in a single switch fail, because the XPS can back up only one of them, it denies power to both power supplies, and the switch shuts down. This occurs only in mixed power mode.

If a switch is connected to a port configured as RPS, but neither of the power supplies is RPS, the RPS port configuration is rejected and the XPS attempts to add the switch to a power stack. If the switch is not capable of operating in SP mode (is not a stackable switch), the port is disabled.

Ports in RPS mode have a configurable priority. The default priority is based on the XPS port number, with port 1 as the highest priority port. A higher priority port has a higher precedence for backup than a lower priority port. If a switch connected to a higher priority port has a power supply failure while a switch connected to a low priority port is being backed up, the XPS drops power to the low priority port to supply power to the high priority port.

Stack Power Mode

Stack-power mode is used only on Catalyst switches in a power stack. With no XPS, a power stack operates in ring topology with a maximum of four switches in the stack. When an XPS is in the power stack, you can connect up to nine switches in the stack plus the XPS, providing power budgets to power stack members similar to stack-power ring topology operation.

All Catalyst switches connected to an XPS on SP ports are part of the same power stack, and all power from the XPS and the switches is shared across all switches in the stack. Power sharing is the default mode, but the XPS supports the same stack power modes that are supported in a ring topology (strict and nonstrict power-sharing or redundant modes).

The XPS uses neighbor discovery to create the power stack. When it discovers a Catalyst switch on an unconfigured port, it marks the port as an SP port, and the switch joins the power stack. The XPS notifies the switch, begins the power-budgeting process, and assigns budgets to each switch in the power stack based on their requirements, priorities, current power allocations, and the stack aggregate power capability.

The XPS sends the power budget to each switch. If not enough input power is available to provide every switch with its maximum requested power, power is distributed based on priority. Switches with the highest priority receive required power first, followed by any powered devices that have already been allocated power, in order of their priority. Any remaining power is distributed equally through the stack.

The RPS port priority (1 through 9) does not affect stack power priority. Each switch participating in stack power has its own system priority and a high and low priority for devices connected to its ports. These priorities are used for stack power, as is the case in a ring topology. You configure stack power priority for the system and for high and low-priority ports by using the **power-priority switch**, **power-priority high**, and **power-priority low** commands in switch stack power configuration mode. If a system or set of powered devices are using the default priority, the XPS automatically assigns a priority (1 through 27), with lower MAC addresses receiving higher priorities.

There are four power stack modes: power sharing, strict power sharing, redundant, or strict redundant. You configure the power stack mode by using the **mode** {**power-sharing** | **redundant**} [**strict**] command in power-stack configuration mode. The **power-sharing** or **redundant** configurations affect the power budgeting

aspect of the stack; **strict** or non-strict affects the actions of the PoE application when a budget reduction does not result in load shedding.

- In power sharing modes (strict or nonstrict), the stack power budget is the cumulative capacity of all the power supplies in the stack (minus 30 W reserved power). This is the default.
- In redundant modes (strict or nonstrict), the stack power budget is the total available power (minus 30 W) after the capacity of the largest power supply in the power stack is subtracted. Redundant mode guarantees that no switch or powered device loses power or experiences load sheds if a single power supply fails, but load sheds can occur if more than one power supply fails.
- In strict modes, if a loss of input power results in reduced power budgets but does not result in any hardware load shedding, the XPS automatically begins denying power to low-priority powered devices and then the high-priority powered devices until the amount of allocated power is less than or equal to the amount of available PoE power.
- In nonstrict modes, in the event of a power reduction, the amount of allocated power is allowed to fall under budget.

For example, a system with a total PoE budget (available power) of 400 W can allocate 390 W of the budget (allocated power) to powered devices. The allocated power of a device is the maximum amount of power that the device needs. The actual power consumption (consumed power) for a set of powered devices is usually not equal to the allocated power. In this example, the actual power might be approximately 200 W. If a power loss in the stack reduces the available power to 210 W, this amount is enough to sustain the power being consumed by the powered devices, but less than the worst-case allocated power, which would put the system *under budget*. In strict mode, the stack would immediately deny power to powered devices until the allocated power was 210 W or less. In nonstrict mode, no action is taken, and the state is allowed to persist. In nonstrict mode if the actual power consumption becomes more than 210 W, this triggers a load shed and can result in the loss of power to all powered devices or switches with the lowest priority level.

Mixed Modes

The XPS 2200 can also operate in mixed mode, where some ports connected to switches are RPS and others are SP. At least one power supply must be an RPS power supply in this configuration. The power supply in the XPS can back up only one switch power supply and the XPS supply must be greater than the largest power supply in a switch connected to an XPS port in RPS mode.

Switches connected to SP ports belong to a single power stack. If the SP switches have a large enough power budget, an SP power supply is not required on the XPS. When an XPS power supply is configured, its power is added to the power pool shared by the power stack.

XPS 2200 System Defaults

The default role for a port is Auto-SP, where the power mode is determined by the switch connected to the port (RPS for Catalyst running the LAN Base image, or SP for Catalyst switches running the IP Base image or IP Services image).

The default for the XPS power supply A (PS1) is RPS mode. The default for power supply B (PS2) is SP mode.

The default mode for all ports and power supplies is enabled.

On ports configured for RPS, the default priority is the same as the port number.

How to Configure the XPS 2200

You can configure the XPS from any switch connected to an XPS port. If you enter XPS configuration commands on more than one switch, the last configuration applied takes effect. Only the switch and port name are saved in the switch configuration file.

Configuring System Names

Procedure

	Command or Action	Purpose
Step 1	**enable**	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal**	Enters global configuration mode.
Step 3	**power xps** *switch-number* **name** {*name* | **serialnumber**}	**Note** The *switch-number* appears only on Catalyst switches and represents the switch number in the data stack, a value from 1 to 9. In a stacked system, the switch-number entered must be the switch number of the active switch. Configures a name for the XPS 2200 system. • *name*—Enter a name for the XPS 2200 system. The name can have up to 20 characters. • **serialnumber**—Use the serial number of the XPS 2200 as the system name.
Step 4	**power xps** *switch-number* **port** {*name* | **hostname** | **serialnumber**}	**Note** The *switch-number* appears only on Catalyst switches and represents the device number in the data stack, a value from 1 to 9. Configures a name for an XPS 2200 port connected to the device. • *name*—Enter a name for the XPS 2200 port. • **serialnumber**—Use the serial number of the device connected to the port. • **hostname**—Use the hostname of the device connected to the port.

	Command or Action	Purpose
Step 5	**end**	Returns to privileged EXEC mode.
Step 6	**show env xps system**	Verifies the configured name of the system and ports.
Step 7	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring XPS Ports

Procedure

	Command or Action	Purpose
Step 1	**enable**	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**power xps** *switch-number* **port** {*number* | **connected**} **mode** {**disable** | **enable**}	**Note** The *switch-number* appears only on Catalyst switches and represents the switch number in the data stack, a value from 1 to 9. Sets the port to be enabled or disabled. • *number*—Enter the XPS 2200 port number. The range is 1 to 9. • **connected**—Enter this keyword if you do not know the port number to which the switch is connected. • **mode disable**—Disable (shut down) the XPS port. **Note** Disabling an XPS port is like removing the cable and appears the same in the **show** command outputs. If the physical cable is connected, you can still use the **enable** keyword to enable the port. • **mode enable**—Enable the XPS port. This is the default.
Step 3	**power xps** *switch-number* **port** {*number* | **connected**} **role** {**auto** | **rps**}	**Note** The *switch-number* represents the switch number in the data stack, a value from 1 to 9. Sets the role of the XPS port.

	Command or Action	Purpose
		• **role auto**—The port mode is determined by the switch connected to the port. This is the default. • **role RPS**—The XPS acts as a back up if the switch power supply fails. At least one RPS power supply must be in RPS mode for this configuration.
Step 4	**power xps** *switch-number* **port** {*number* | **connected**} **priority** *port-priority*	**Note** The *switch-number* represents the switch number in the data stack, a value from 1 to 9. Sets the RPS priority of the port, where higher priority ports take precedence over low priority ports if multiple power supplies fail. This command takes effect only when the port mode is RPS. When the port mode is stack power, you set priority by using the stack power commands. • **priority** *port-priority*—Set the RPS priority of the port. The range is 1 to 9, with 1 being the highest priority. The default priority is the XPS port number.
Step 5	**show env xps port**	Verifies the XPS configuration of the port.
Step 6	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring XPS Power Supplies

Procedure

	Command or Action	Purpose
Step 1	**enable**	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**power xps** *switch-number* **supply** {**A** | **B**} **mode** {**rps** | **sp**}	**Note** The *switch-number* represents the switch number in the data stack, a value from 1 to 9. Sets the XPS power supply mode. • **supply** {**A** | **B**}—Select the power supply to configure. Power supply A is on the left (labeled PS1) and power supply B (PS2) is on the right.

	Command or Action	Purpose
		• **mode rps**—Set the power supply mode to RPS, to back up connected switches. This is the default setting for power supply A (PS1). • **mode sp**—Set the power supply mode to stack power (SP), to participate in the power stack. This is the default setting for power supply B (PS2).
Step 3	**power xps** *switch-number* **supply** {**A** | **B**} {**on** | **off**}	**Note** The *switch-number* represents the switch number in the data stack, a value from 1 to 9. Sets the XPS power supply to be on or off. The default is for both power supplies to be on.
Step 4	**end**	Returns to privileged EXEC mode.
Step 5	**show env xps power**	Displays the status of the XPS power supplies.

Monitoring and Maintaining the XPS 2200

Command	Purpose
show env xps system	Verifies the configured name of the system and ports.
show env xps port	Verifies the XPS configuration of the port.
show env xps power	Displays the status of the XPS power supplies.

Additional References for XPS 2200

Related Documents

Related Topic	Document Title
For complete syntax and usage information for the commands used in this chapter.	See the *Interface and Hardware Commands* section of the *Command Reference (Catalyst 3850 Series).*

Standards and RFCs

Standard/RFC	Title
None	--

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

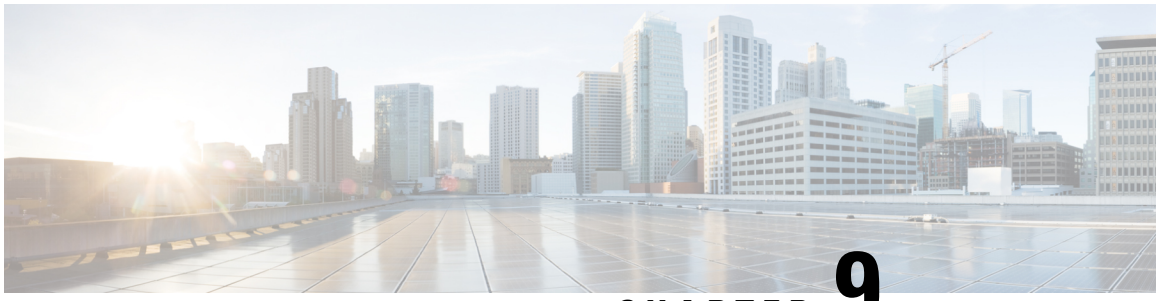
Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information for XPS 2200

Table 12: Feature Information for XPS 2200

Release	Feature
Cisco IOS XE 15.2.(3)E1	XPS 2200 feature is introduced



CHAPTER 9

Configuring EEE

- Restrictions for EEE, on page 93
- Information About EEE, on page 93
- How to Configure EEE, on page 94
- Monitoring EEE, on page 95
- Configuration Examples for Configuring EEE, on page 96
- Additional References for EEE, on page 96
- Feature Information for Configuring EEE, on page 96

Restrictions for EEE

EEE has the following restrictions:

- Changing the EEE configuration resets the interface because the device has to restart Layer 1 autonegotiation.
- You might want to enable the Link Layer Discovery Protocol (LLDP) for devices that require longer wakeup times before they are able to accept data on their receive paths. Doing so enables the device to negotiate for extended system wakeup times from the transmitting link partner.
- EEE is not supported on TenGigabit Ethernet and Multigigabit Ethernet ports.

Information About EEE

EEE Overview

Energy Efficient Ethernet (EEE) is an IEEE 802.3az standard that is designed to reduce power consumption in Ethernet networks during idle periods.

Default EEE Configuration

EEE is disabled by default.

How to Configure EEE

You can enable or disable EEE on an interface that is connected to an EEE-capable link partner.

Enabling or Disabling EEE

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Device(config)# interface gigabitethernet 1/0/1`	Specifies the interface to be configured, and enter interface configuration mode.
Step 3	**power efficient-ethernet auto** **Example:** `Device(config-if)# power efficient-ethernet auto`	Enables EEE on the specified interface. When EEE is enabled, the device advertises and autonegotiates EEE to its link partner.
Step 4	**no power efficient-ethernet auto** **Example:** `Device(config-if)# no power efficient-ethernet auto`	Disables EEE on the specified interface.
Step 5	**end** **Example:** `Device(config-if)# end`	Returns to privileged EXEC mode.
Step 6	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Monitoring EEE

Table 13: Commands for Displaying EEE Settings

Command	Purpose
show eee capabilities interface *interface-id*	Displays EEE capabilities for the specified interface.
show eee status interface *interface-id*	Displays EEE status information for the specified interface.
show eee counters interface *interface-id*	Displays EEE counters for the specified interface.

Following are examples of the **show eee** commands

```
Switch#show eee capabilities interface gigabitEthernet2/0/1
Gi2/0/1
EEE(efficient-ethernet): yes (100-Tx and 1000T auto)
Link Partner : yes (100-Tx and 1000T auto)

ASIC/Interface : EEE Capable/EEE Enabled
```

```
Switch#show eee status interface gigabitEthernet2/0/1
Gi2/0/1 is up
EEE(efficient-ethernet): Operational
Rx LPI Status : Low Power
Tx LPI Status : Low Power
Wake Error Count : 0

ASIC EEE STATUS
Rx LPI Status : Receiving LPI
Tx LPI Status : Transmitting LPI
Link Fault Status : Link Up
Sync Status : Code group synchronization with data stream intact
```

```
Switch#show eee counters interface gigabitEthernet2/0/1

LP Active Tx Time (10us) : 66649648
LP Transitioning Tx : 462
LP Active Rx Time (10us) : 64911682
LP Transitioning Rx : 153
```

Examples for Cataylst Digital Building Series Switches

```
Switch#show eee capabilities interface gig1/0/1
Gi1/0/1
EEE(efficient-ethernet): yes (100-Tx and 1000T auto)
Link Partner : no
```

```
Switch#show eee status int gig1/0/1
Gi1/0/1 is up
EEE(efficient-ethernet): Disagreed
Rx LPI Status : None
Tx LPI Status : None
Wake Error Count : 0
```

Configuration Examples for Configuring EEE

This example shows how to enable EEE for an interface:

```
Device# configure terminal
Device(config)# interface gigabitethernet 1/0/1
Device(config-if)# power efficient-ethernet auto
```

This example shows how to disable EEE for an interface:

```
Device# configure terminal
Device(config)# interface gigabitethernet 1/0/1
Device(config-if)# no power efficient-ethernet auto
```

Additional References for EEE

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature Information for Configuring EEE

Release	Modification
Cisco IOS XE 3.2SE	This feature was introduced.