



# Configuring Ethernet Interfaces

This module describes the configuration of Ethernet interfaces.

The distributed Gigabit Ethernet, 10-Gigabit, 40-Gigabit, 100-Gigabit Ethernet, and Fast Ethernet architecture and features deliver network scalability and performance, while enabling service providers to offer high-density, high-bandwidth networking solutions designed to interconnect the router with other systems in POPs, including core and edge routers and Layer 2 switches.



**Tip** You can programmatically configure and manage the Ethernet interfaces using `openconfig-ethernet-if.yang` and `openconfig-interfaces.yang` OpenConfig data models. To get started with using data models, see the *Programmability Configuration Guide for Cisco CRS Routers*.

## Feature History for Configuring Ethernet Interfaces

Release	Modification
Release 3.0	This feature was introduced on the Cisco CRS-1 Router.
Release 3.2	Support was added for the Cisco CRS-1 Router for the SIP-800. Support for the 8-Port Gigabit Ethernet SPA was introduced on the Cisco CRS-1 Router.
Release 3.3.0	Support was added for egress MAC accounting on the 8-port 10-Gigabit Ethernet PLIM.
Release 3.4.0	The Layer 2 Virtual Private Network (L2VPN) feature was first supported on Ethernet interfaces. Support was added for the 8-Port 1-Gigabit Ethernet SPA.
Release 3.5.0	Support was added on the Cisco CRS-1 Router for the 1-port 10-Gigabit Ethernet WAN SPA.

Release 4.0.0	Support for the following physical layer interface modules (PLIMs) was added: <ul style="list-style-type: none"> <li>• 14-Port 10-Gigabit Ethernet LAN/WAN-PHY PLIM (14X10GBE-WL-XFP) (with the Cisco CRS-3 Modular Services Card or Cisco CRS-3 Forwarding Processor Card)</li> <li>• 20-Port 10-Gigabit Ethernet LAN/WAN-PHY PLIM (20X10GBE-WL-XFP) (with the Cisco CRS-3 Modular Services Card or Cisco CRS-3 Forwarding Processor Card)</li> </ul>
Release 4.0.1	Support for the following PLIMs was added: <ul style="list-style-type: none"> <li>• 1-Port 100-Gigabit Ethernet PLIM (1X100GBE) (with the Cisco CRS-3 Modular Services Card or Cisco CRS-3 Forwarding Processor Card)</li> <li>• 4-Port 10-Gigabit Ethernet LAN/WAN-PHY PLIM (4-10GBE-WL-XFP)</li> <li>• 8-Port 10-Gigabit Ethernet LAN/WAN-PHY PLIM (8-10GBE-WL-XFP)</li> </ul>
Release 4.2.3	Support for Link Layer Discovery Protocol (LLDP) was added.

- [Configuring Ethernet Interfaces, on page 2](#)
- [Prerequisites for Configuring Ethernet Interfaces, on page 3](#)
- [Information About Configuring Ethernet, on page 4](#)
- [How to Configure Ethernet, on page 15](#)
- [Configuration Examples for Ethernet, on page 34](#)
- [How to Configure Interfaces in Breakout Mode, on page 36](#)

## Configuring Ethernet Interfaces

This module describes the configuration of Ethernet interfaces.

The distributed Gigabit Ethernet, 10-Gigabit, 40-Gigabit, 100-Gigabit Ethernet, and Fast Ethernet architecture and features deliver network scalability and performance, while enabling service providers to offer high-density, high-bandwidth networking solutions designed to interconnect the router with other systems in POPs, including core and edge routers and Layer 2 switches.



**Tip** You can programmatically configure and manage the Ethernet interfaces using `openconfig-ethernet-if.yang` and `openconfig-interfaces.yang` OpenConfig data models. To get started with using data models, see the *Programmability Configuration Guide for Cisco CRS Routers*.

### Feature History for Configuring Ethernet Interfaces

Release	Modification
Release 3.0	This feature was introduced on the Cisco CRS-1 Router.

Release 3.2	Support was added for the Cisco CRS-1 Router for the SIP-800. Support for the 8-Port Gigabit Ethernet SPA was introduced on the Cisco CRS-1 Router.
Release 3.3.0	Support was added for egress MAC accounting on the 8-port 10-Gigabit Ethernet PLIM.
Release 3.4.0	The Layer 2 Virtual Private Network (L2VPN) feature was first supported on Ethernet interfaces. Support was added for the 8-Port 1-Gigabit Ethernet SPA.
Release 3.5.0	Support was added on the Cisco CRS-1 Router for the 1-port 10-Gigabit Ethernet WAN SPA.
Release 4.0.0	Support for the following physical layer interface modules (PLIMs) was added: <ul style="list-style-type: none"> <li>• 14-Port 10-Gigabit Ethernet LAN/WAN-PHY PLIM (14X10GBE-WL-XFP) (with the Cisco CRS-3 Modular Services Card or Cisco CRS-3 Forwarding Processor Card)</li> <li>• 20-Port 10-Gigabit Ethernet LAN/WAN-PHY PLIM (20X10GBE-WL-XFP) (with the Cisco CRS-3 Modular Services Card or Cisco CRS-3 Forwarding Processor Card)</li> </ul>
Release 4.0.1	Support for the following PLIMs was added: <ul style="list-style-type: none"> <li>• 1-Port 100-Gigabit Ethernet PLIM (1X100GBE) (with the Cisco CRS-3 Modular Services Card or Cisco CRS-3 Forwarding Processor Card)</li> <li>• 4-Port 10-Gigabit Ethernet LAN/WAN-PHY PLIM (4-10GBE-WL-XFP)</li> <li>• 8-Port 10-Gigabit Ethernet LAN/WAN-PHY PLIM (8-10GBE-WL-XFP)</li> </ul>
Release 4.2.3	Support for Link Layer Discovery Protocol (LLDP) was added.

## Prerequisites for Configuring Ethernet Interfaces

You must be in a user group associated with a task group that includes the proper task IDs. The command reference guides include the task IDs required for each command. If you suspect user group assignment is preventing you from using a command, contact your AAA administrator for assistance.

Before configuring Ethernet interfaces, be sure that these tasks and conditions are met:

- Confirm that at least one of these line cards supported on the router is installed:
  - 8-Port Fast Ethernet SPA
  - 4-Port 1-Gigabit Ethernet physical layer interface module (PLIM)
  - 2-Port Gigabit Ethernet SPA
  - 5-Port Gigabit Ethernet SPA

- 8-Port Gigabit Ethernet SPA
  - 10-Port Gigabit Ethernet SPA
  - 1-Port 10-Gigabit Ethernet SPA
  - 1-Port 10-Gigabit Ethernet WAN SPA
  - 4-Port 10-Gigabit Ethernet LAN/WAN-PHY PLIM (4-10GBE-WL-XFP)
  - 8-Port 10-Gigabit Ethernet LAN/WAN-PHY PLIM (8-10GBE-WL-XFP)
  - 8-Port 10-Gigabit Ethernet PLIM
  - 14-Port 10-Gigabit Ethernet LAN/WAN-PHY PLIM (14X10GBE-WL-XFP) (with the Cisco CRS-3 Modular Services Card or Cisco CRS-3 Forwarding Processor Card)
  - 20-Port 10-Gigabit Ethernet LAN/WAN-PHY PLIM (20X10GBE-WL-XFP) (with the Cisco CRS-3 Modular Services Card or Cisco CRS-3 Forwarding Processor Card)
  - 1-Port 100-Gigabit Ethernet PLIM (1X100GBE) (with the Cisco CRS-3 Modular Services Card or Cisco CRS-3 Forwarding Processor Card)
- Know the interface IP address.
  - You know how to apply the specify the generalized interface name with the generalized notation *rack/slot/module/port*.
  - If you are configuring a 10-Gigabit Ethernet interface on a 10-GE DWDM PLIM, you must have configured the DWDM controller.

## Information About Configuring Ethernet

Ethernet is defined by the IEEE 802.3 international standard. It enables the connection of up to 1024 nodes over coaxial, twisted-pair, or fiber-optic cable.

The Cisco CRS-1 Router supports Gigabit Ethernet (1000 Mbps), 10-Gigabit Ethernet (10 Gbps), and 100-Gigabit Ethernet (100 Gbps) interfaces.

This section provides the following information sections:

### Default Configuration Values for Gigabit Ethernet and 10-Gigabit Ethernet

This table describes the default interface configuration parameters that are present when an interface is enabled on a Gigabit Ethernet or 10-Gigabit Ethernet modular services card and its associated PLIM.




---

**Note** You must use the **shutdown** command to bring an interface administratively down. The interface default is **no shutdown**. When a modular services card is first inserted into the router, if there is no established preconfiguration for it, the configuration manager adds a shutdown item to its configuration. This shutdown can be removed only by entering the **no shutdown** command.

---

**Table 1: Gigabit Ethernet and 10-Gigabit Ethernet Modular Services Card Default Configuration Values**

Parameter	Configuration File Entry	Default Value
MAC accounting	<b>mac-accounting</b>	<b>off</b>
Flow control	<b>flow-control</b>	egress on ingress off
MTU	<b>mtu</b>	<ul style="list-style-type: none"> <li>• 1514 bytes for normal frames</li> <li>• 1518 bytes for 802.1Q tagged frames.</li> <li>• 1522 bytes for Q-in-Q frames.</li> </ul>
MAC address	<b>mac address</b>	Hardware burned-in address (BIA)

## Default Configuration Values for Fast Ethernet

This table describes the default interface configuration parameters that are present when an interface is enabled on the Fast Ethernet SPA card and its associated PLIM.



**Note** You must specifically configure the **shutdown** command to bring an interface administratively down. The interface default is **no shutdown**. When a modular services card is first inserted into the router, if there is no established preconfiguration for it, the configuration manager adds a shutdown item to its configuration. This shutdown can be removed only by entering the **no shutdown** command.

**Table 2: Fast Ethernet Default Configuration Values**

Parameter	Configuration File Entry	Default Value
MAC accounting	<b>mac-accounting</b>	<b>off</b>
Duplex operation	<b>duplex full</b> <b>duplex half</b>	Auto-negotiates duplex operation
MTU	<b>mtu</b>	1500 bytes
Interface speed	<b>speed</b>	100 Mbps
Auto-negotiation	<b>negotiation auto</b>	<b>disable</b>

## Layer 2 VPN on Ethernet Interfaces

Layer 2 Virtual Private Network (L2VPN) connections emulate the behavior of a LAN across an L2 switched, IP or MPLS-enabled IP network, allowing Ethernet devices to communicate with each other as if they were connected to a common LAN segment.

The L2VPN feature enables service providers (SPs) to provide Layer 2 services to geographically disparate customer sites. Typically, an SP uses an access network to connect the customer to the core network. This access network may use a mixture of Layer 2 technologies, such as Ethernet, ATM and Frame Relay. The connection between the customer site and the nearby SP edge router is known as an Attachment Circuit (AC).

Traffic from the customer travels over this link to the edge of the SP core network. The traffic then tunnels through an L2VPN over the SP core network to another edge router. The edge router sends the traffic down another attachment circuit (AC) to the customer's remote site.

The L2VPN feature enables users to implement different types of end-to-end services.

Cisco IOS XR Software supports a point-to-point end-to-end service, where two Ethernet circuits are connected together. An L2VPN Ethernet port can operate in one of two modes:

- **Port Mode**—In this mode, all packets reaching the port are sent over the PW (pseudowire), regardless of any VLAN tags that are present on the packets. In VLAN mode, the configuration is performed under the l2transport configuration mode.
- **VLAN Mode**—Each VLAN on a CE (customer edge) or access network to PE (provider edge) link can be configured as a separate L2VPN connection (using either VC type 4 or VC type 5). In VLAN mode, the configuration is performed under the individual subinterface.




---

**Note** The system sets a limit of 24K single vlan tags per NP and a 64K LC limit on the following line cards:

- A9K-MOD400-SE
  - A9K-MOD400-CM
  - A9K-MOD200-SE/CM
  - Cisco ASR 9000 Series 24-port and 48-port dual-rate 10GE/1GE SE/CM
  - A9K-8x100 SE/CM
  - A99-8x100 SE/CM
- 

Switching can take place in three ways:

- **AC-to-PW**—Traffic reaching the PE is tunneled over a PW (and conversely, traffic arriving over the PW is sent out over the AC). This is the most common scenario.
- **Local switching**—Traffic arriving on one AC is immediately sent out of another AC without passing through a pseudowire.
- **PW stitching**—Traffic arriving on a PW is not sent to an AC, but is sent back into the core over another PW.

Keep the following in mind when configuring L2VPN on an Ethernet interface:

- L2VPN links support QoS (Quality of Service) and MTU (maximum transmission unit) configuration.
- If your network requires that packets are transported transparently, you may need to modify the packet's destination MAC (Media Access Control) address at the edge of the Service Provider (SP) network. This prevents the packet from being consumed by the devices in the ST network.
- Cisco IOS XR software supports up to 4,000 ACs per line card. Note that not all line cards can support as many as 4,000 ACs. Refer to the specifications of the individual line card for details on the maximum number of ACs supported.

Use the **show interfaces** command to display AC and PW information.

To configure a point-to-point pseudowire xconnect on an AC, see the *Implementing MPLS Layer 2 VPNs* module of the *Cisco IOS XR Multiprotocol Label Switching Configuration Guide*.

To attach Layer 2 service policies, such as QoS, to the Ethernet interface, refer to the appropriate Cisco IOS XR software configuration guide.

## Gigabit Ethernet Protocol Standards Overview

The Gigabit Ethernet interfaces support the following protocol standards:

These standards are further described in the sections that follow.

### IEEE 802.3 Physical Ethernet Infrastructure

The IEEE 802.3 protocol standards define the physical layer and MAC sublayer of the data link layer of wired Ethernet. IEEE 802.3 uses Carrier Sense Multiple Access with Collision Detection (CSMA/CD) access at a variety of speeds over a variety of physical media. The IEEE 802.3 standard covers 10 Mbps Ethernet. Extensions to the IEEE 802.3 standard specify implementations for Gigabit Ethernet, 10-Gigabit Ethernet, and Fast Ethernet.

### IEEE 802.3ab 1000BASE-T Gigabit Ethernet

The IEEE 802.3ab protocol standards, or Gigabit Ethernet over copper (also known as 1000BaseT) is an extension of the existing Fast Ethernet standard. It specifies Gigabit Ethernet operation over the Category 5e/6 cabling systems already installed, making it a highly cost-effective solution. As a result, most copper-based environments that run Fast Ethernet can also run Gigabit Ethernet over the existing network infrastructure to dramatically boost network performance for demanding applications.

### IEEE 802.3z 1000 Mbps Gigabit Ethernet

Gigabit Ethernet builds on top of the Ethernet protocol, but increases speed tenfold over Fast Ethernet to 1000 Mbps, or 1 Gbps. Gigabit Ethernet allows Ethernet to scale from 10 or 100 Mbps at the desktop to 100 Mbps up to 1000 Mbps in the data center. Gigabit Ethernet conforms to the IEEE 802.3z protocol standard.

By leveraging the current Ethernet standard and the installed base of Ethernet and Fast Ethernet switches and routers, network managers do not need to retrain and relearn a new technology in order to provide support for Gigabit Ethernet.

### IEEE 802.3ae 10 Gbps Ethernet

Under the International Standards Organization's Open Systems Interconnection (OSI) model, Ethernet is fundamentally a Layer 2 protocol. 10-Gigabit Ethernet uses the IEEE 802.3 Ethernet MAC protocol, the IEEE

802.3 Ethernet frame format, and the minimum and maximum IEEE 802.3 frame size. 10 Gbps Ethernet conforms to the IEEE 802.3ae protocol standards.

Just as 1000BASE-X and 1000BASE-T (Gigabit Ethernet) remained true to the Ethernet model, 10-Gigabit Ethernet continues the natural evolution of Ethernet in speed and distance. Because it is a full-duplex only and fiber-only technology, it does not need the carrier-sensing multiple-access with the CSMA/CD protocol that defines slower, half-duplex Ethernet technologies. In every other respect, 10-Gigabit Ethernet remains true to the original Ethernet model.

## IEEE 802.3ba 100 Gbps Ethernet

IEEE 802.3ba is supported on the Cisco 1-Port 100-Gigabit Ethernet PLIM beginning in Cisco IOS XR 4.0.1.

## MAC Address

A MAC address is a unique 6-byte address that identifies the interface at Layer 2.

## MAC Accounting

The MAC address accounting feature provides accounting information for IP traffic based on the source and destination MAC addresses on LAN interfaces. This feature calculates the total packet and byte counts for a LAN interface that receives or sends IP packets to or from a unique MAC address. It also records a time stamp for the last packet received or sent.

These statistics are used for traffic monitoring, debugging and billing. For example, with this feature you can determine the volume of traffic that is being sent to and/or received from various peers at NAPS/peering points. This feature is currently supported on Ethernet, FastEthernet, and bundle interfaces and supports Cisco Express Forwarding (CEF), distributed CEF (dCEF), flow, and optimum switching.



---

**Note** A maximum of 512 MAC addresses per trunk interface are supported for MAC address accounting.

---

## Ethernet MTU

The Ethernet maximum transmission unit (MTU) is the size of the largest frame, minus the 4-byte frame check sequence (FCS), that can be transmitted on the Ethernet network. Every physical network along the destination of a packet can have a different MTU.

Cisco IOS XR software supports two types of frame forwarding processes:

- Fragmentation for IPv4 packets—In this process, IPv4 packets are fragmented as necessary to fit within the MTU of the next-hop physical network.



---

**Note** IPv6 does not support fragmentation.

---

- MTU discovery process determines largest packet that size—This process is available for all IPv6 devices, and for originating IPv4 devices. In this process, the originating IP device determines the size of the largest IPv6 or IPv4 packet that can be sent without being fragmented. The largest packet is equal to the



smallest MTU of any network between the IP source and the IP destination devices. If a packet is larger than the smallest MTU of all the networks in its path, that packet is fragmented as necessary. This process ensures that the originating device does not send an IP packet that is too large.

Jumbo frame support is automatically enabled for frames that exceed the standard frame size. The default value is 1514 for standard frames and 1518 for 802.1Q tagged frames. These numbers exclude the 4-byte frame check sequence (FCS).

## Flow Control on Ethernet Interfaces

The flow control used on 10-Gigabit Ethernet interfaces consists of periodically sending flow control pause frames. It is fundamentally different from the usual full- and half-duplex flow control used on standard management interfaces. Flow control can be activated or deactivated for ingress traffic only. It is automatically implemented for egress traffic.

## 802.1Q VLAN

A VLAN is a group of devices on one or more LANs that are configured so that they can communicate as if they were attached to the same wire, when in fact they are located on a number of different LAN segments. Because VLANs are based on logical instead of physical connections, it is very flexible for user and host management, bandwidth allocation, and resource optimization.

The IEEE's 802.1Q protocol standard addresses the problem of breaking large networks into smaller parts so broadcast and multicast traffic does not consume more bandwidth than necessary. The standard also helps provide a higher level of security between segments of internal networks.

The 802.1Q specification establishes a standard method for inserting VLAN membership information into Ethernet frames.

## VRRP

The Virtual Router Redundancy Protocol (VRRP) eliminates the single point of failure inherent in the static default routed environment. VRRP specifies an election protocol that dynamically assigns responsibility for a virtual router to one of the VPN concentrators on a LAN. The VRRP VPN concentrator controlling the IP addresses associated with a virtual router is termed as the primary concentrator, and forwards packets sent to those IP addresses. When the primary concentrator becomes unavailable, a backup VPN concentrator takes over.

For more information on VRRP, see the *Implementing VRRP on Cisco IOS XR Software* module of .

## HSRP

Hot Standby Routing Protocol (HSRP) is a proprietary protocol from Cisco. HSRP is a routing protocol that provides backup to a router in the event of failure. Several routers are connected to the same segment of an Ethernet, FDDI, or token-ring network and work together to present the appearance of a single virtual router on the LAN. The routers share the same IP and MAC addresses and therefore, in the event of failure of one router, the hosts on the LAN are able to continue forwarding packets to a consistent IP and MAC address. The transfer of routing responsibilities from one device to another is transparent to the user.

HSRP is designed to support non disruptive switchover of IP traffic in certain circumstances and to allow hosts to appear to use a single router and to maintain connectivity even if the actual first hop router they are

using fails. In other words, HSRP protects against the failure of the first hop router when the source host cannot learn the IP address of the first hop router dynamically. Multiple routers participate in HSRP and in concert create the illusion of a single virtual router. HSRP ensures that one and only one of the routers is forwarding packets on behalf of the virtual router. End hosts forward their packets to the virtual router.

The router forwarding packets is known as the *active router*. A standby router is selected to replace the active router should it fail. HSRP provides a mechanism for determining active and standby routers, using the IP addresses on the participating routers. If an active router fails a standby router can take over without a major interruption in the host's connectivity.

HSRP runs on top of User Datagram Protocol (UDP), and uses port number 1985. Routers use their actual IP address as the source address for protocol packets, not the virtual IP address, so that the HSRP routers can identify each other.

For more information on HSRP, see the *Implementing HSRP on Cisco IOS XR Software* module of *IP Addresses and Services Configuration Guide*.

## Duplex Mode on Fast Ethernet Interfaces

Fast Ethernet ports support the duplex transmission type. Full-duplex mode enables the simultaneous data transmission between a sending station and a receiving station, while half-duplex mode enables data transmission in only one direction at a time.

When configuring duplex mode on a Fast Ethernet interface, keep the following in mind:

- If auto-negotiation is enabled on the interface, the default is duplex negotiated.
- If auto-negotiation is disabled on the interface, the default is full-duplex.



---

**Note** You can configure duplex mode on Fast Ethernet interfaces only. Gigabit Ethernet and 10-Gigabit Ethernet interfaces always run in full-duplex mode.

---

## Fast Ethernet Interface Speed

You can configure the interface speed on Fast Ethernet interfaces. Keep the following in mind when configuring the speed for a Fast Ethernet interface:

- If auto-negotiation is enabled on an interface, the default is speed negotiated.
- If auto-negotiation is disabled on an interface, the default speed is the maximum speed allowed on the interface.



---

**Note** Both ends of a link must have the same interface speed. A manually configured interface speed overrides any auto-negotiated speed, which can prevent a link from coming up if the configured interface speed at one end of a link is different from the interface speed on the other end.

---

## Link Autonegotiation on Ethernet Interfaces

Link autonegotiation ensures that devices that share a link segment are automatically configured with the highest performance mode of interoperation. Use the **negotiation auto** command in interface configuration mode to enable link autonegotiation on an Ethernet interface. On line card Ethernet interfaces, link autonegotiation is disabled by default.



**Note** The **negotiation auto** command is available on Gigabit Ethernet and Fast Ethernet interfaces only.

This table describes the performance of the system for different combinations of the duplex and speed modes. The specified **duplex** command configured with the specified **speed** command produces the resulting system action, provided that you have configured autonegotiation on the interface.

*Table 3: Relationship Between duplex and speed Commands*

duplex Command	speed Command	
<b>no duplex</b>	<b>no speed</b>	Auto-negotiates both speed and duplex modes.
<b>no duplex</b>	<b>speed 1000</b>	Auto-negotiates for duplex mode and forces 1000 Mbps.
<b>no duplex</b>	<b>speed 100</b>	Auto-negotiates for duplex mode and forces 100 Mbps.
<b>no duplex</b>	<b>speed 10</b>	Auto-negotiates for duplex mode and forces 10 Mbps.
<b>full-duplex</b>	<b>no speed</b>	Forces full duplex and auto-negotiates for speed.
<b>full-duplex</b>	<b>speed 1000</b>	Forces full duplex and 1000 Mbps.
<b>full-duplex</b>	<b>speed 100</b>	Forces full duplex and 100 Mbps.
<b>full-duplex</b>	<b>speed 10</b>	Forces full duplex and 10 Mbps.
<b>half-duplex</b>	<b>no speed</b>	Forces half duplex and auto-negotiates for speed.
<b>half-duplex</b>	<b>speed 1000</b>	Forces half duplex and 1000 Mbps.
<b>half-duplex</b>	<b>speed 100</b>	Forces half duplex and 100 Mbps.
<b>half-duplex</b>	<b>speed 10</b>	Forces half duplex and 10 Mbps.

## LLDP



**Note** LLDP is not supported on the FP-X line cards.

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, and switches). 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 Cisco CRS Router also supports the IEEE 802.1AB LLDP. LLDP is also 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 supports a set of attributes that it uses to learn information about neighbor devices. These attributes have a defined format known as a Type-Length-Value (TLV). LLDP supported devices can use TLVs to receive and send information to their neighbors. Details such as configuration information, device capabilities, and device identity can be advertised using this protocol.

In addition to the mandatory TLVs (Chassis ID, Port ID, and Time-to-Live), the router also supports the following basic management TLVs, which are optional:

- Port Description
- System Name
- System Description
- System Capabilities
- Management Address

These optional TLVs are automatically sent when LLDP is active, but you can disable them as needed using the **lldp tlv-select disable** command.

## LLDP Frame Format

LLDP frames use the IEEE 802.3 format, which consists of the following fields:

- Destination address (6 bytes)—Uses a multicast address of 01-80-C2-00-00-0E.
- Source address (6 bytes)—MAC address of the sending device or port.
- LLDP Ethertype (2 bytes)—Uses 88-CC.
- LLDP PDU (1500 bytes)—LLDP payload consisting of TLVs.
- FCS (4 bytes)—Cyclic Redundancy Check (CRC) for error checking.

## LLDP TLV Format

LLDP TLVs carry the information about neighboring devices within the LLDP PDU using the following basic format:

- TLV Header (16 bits), which includes the following fields:
  - TLV Type (7 bits)
  - TLV Information String Length (9 bits)
- TLV Information String (0 to 511 bytes)

## LLDP Operation

LLDP is a one-way protocol. The basic operation of LLDP consists of a device enabled for transmit of LLDP information sending periodic advertisements of information in LLDP frames to a receiving device.

Devices are identified using a combination of the Chassis ID and Port ID TLVs to create an MSAP (MAC Service Access Point). The receiving device saves the information about a neighbor for a certain amount time specified in the TTL TLV, before aging and removing the information.

LLDP supports the following additional operational characteristics:

- LLDP can operate independently in transmit or receive modes.
- LLDP operates as a slow protocol using only untagged frames, with transmission speeds of less than 5 frames per second.
- LLDP packets are sent when the following occurs:
  - The packet update frequency specified by the **lldp timer** command is reached. The default is 30 seconds.
  - When a change in the values of the managed objects occurs from the local system's LLDP MIB.
  - When LLDP is activated on an interface (3 frames are sent upon activation similar to CDP).
- When an LLDP frame is received, the LLDP remote services and PTOPO MIBs are updated with the information in the TLVs.
- LLDP supports the following actions on these TLV characteristics:
  - Interprets a TTL value of 0 as a request to automatically purge the information of the transmitting device. These shutdown LLDPDUs are typically sent prior to a port becoming inoperable.
  - An LLDP frame with a malformed mandatory TLV is dropped.
  - A TLV with an invalid value is ignored.
  - A copy of an unknown organizationally-specific TLV is maintained if the TTL is non-zero, for later access through network management.

## Supported LLDP Functions

The Cisco CRS Router supports the following LLDP functions:

- IPv4 and IPv6 management addresses—In general, both IPv4 and IPv6 addresses will be advertised if they are available, and preference is given to the address that is configured on the transmitting interface.

If the transmitting interface does not have a configured address, then the TLV will be populated with an address from another interface. The advertised LLDP IP address is implemented according to the following priority order of IP addresses for interfaces on the Cisco CRS Router:

- Locally configured address
- MgmtEth0/RP0/CPU0/0
- MgmtEth0/RP0/CPU0/1
- MgmtEth0/RP1/CPU0/0

- MgmtEth0/RP1/CPU0/1
- Loopback interfaces

There are some differences between IPv4 and IPv6 address management in LLDP:

- For IPv4, as long as the IPv4 address is configured on an interface, it can be used as an LLDP management address.
- For IPv6, after the IPv6 address is configured on an interface, the interface status must be Up and pass the DAD (Duplicate Address Detection) process before it can be used as an LLDP management address.
- LLDP is supported for the nearest physically attached, non-tunneled neighbors.
- Port ID TLVs are supported for Ethernet interfaces, subinterfaces, bundle interfaces, and bundle subinterfaces.

## Unsupported LLDP Functions

The following LLDP functions are not supported on the Cisco CRS Router:

- LLDP-MED organizationally unique extension—However, interoperability still exists between other devices that do support this extension.
- Tunneled neighbors, or neighbors more than one hop away.
- LLDP TLVs cannot be disabled on a per-interface basis; However, certain optional TLVs can be disabled globally.
- LLDP SNMP trap `lldpRemTablesChange`.

## Carrier Delay on Ethernet Interfaces

When enabled on an Ethernet interface, the Carrier Delay feature slows the response of the system to line-up or line-down events. You can configure both Carrier Delay up and Carrier Delay down on an interface at the same time.

Carrier Delay up suppresses short line flaps where the line is down, then comes up, then goes down again. A line that was previously down must be up longer than the duration specified for the **carrier-delay up** command before the system is informed that the interface has come up. All flaps that are shorter than the duration specified for the **carrier-delay up** command are suppressed.

Configuring Carrier Delay up helps to ensure that a line is reasonably stable before the system is informed that the interface is up and ready to forward traffic.

Carrier Delay down suppresses short line flaps where the line is up, then goes down, then comes up again. A line that was previously up must be down longer than the duration specified for the **carrier-delay down** command before the system is informed that the interface has gone down. All flaps that are shorter than the value specified for the **carrier-delay down** command are suppressed.

Configuring Carrier Delay down can be beneficial in suppressing very short link flaps, thereby preventing interface flaps. Alternatively, configuring this feature can be beneficial in allowing other line protection equipment to have enough time to intervene.

# How to Configure Ethernet

This section provides the following configuration procedures:

## Configuring Ethernet Interfaces

This section provides the following configuration procedures:

### Configuring Gigabit Ethernet Interfaces

Use the following procedure to create a basic Gigabit Ethernet, 10-Gigabit Ethernet, or 100-Gigabit Ethernet interface configuration.

#### SUMMARY STEPS

1. **show version**
2. **show interfaces** [**GigabitEthernet** | **TenGigE** | **HundredGigE**] *interface-path-id*
3. **configure**
4. **interface** [**GigabitEthernet** | **TenGigE** | **HundredGigE**] *interface-path-id*
5. **ipv4 address** *ip-address mask*
6. **flow-control** {**bidirectional** | **egress** | **ingress**}
7. **mtu** *bytes*
8. **mac-address** *value1.value2.value3*
9. **negotiation auto**
10. **no shutdown**
11. **end** or **commit**
12. **show interfaces** [**GigabitEthernet** | **TenGigE**] *interface-path-id*

#### DETAILED STEPS

	Command or Action	Purpose
<b>Step 1</b>	<b>show version</b> <b>Example:</b> <pre>RP/0/RP0/CPU0:router# show version</pre>	(Optional) Displays the current software version, and can also be used to confirm that the router recognizes the modular services card.
<b>Step 2</b>	<b>show interfaces</b> [ <b>GigabitEthernet</b>   <b>TenGigE</b>   <b>HundredGigE</b> ] <i>interface-path-id</i> <b>Example:</b> <pre>RP/0/RP0/CPU0:router# show interfaces TenGigE 0/1/0/0</pre>	(Optional) Displays the configured interface and checks the status of each interface port. Possible interface types for this procedure are: <ul style="list-style-type: none"> <li>• GigabitEthernet</li> <li>• HundredGigE</li> <li>• TenGigE</li> </ul>

	Command or Action	Purpose
<b>Step 3</b>	<b>configure</b> <b>Example:</b> <pre>RP/0/RP0/CPU0:router# configure terminal</pre>	Enters global configuration mode.
<b>Step 4</b>	<b>interface [GigabitEthernet   TenGigE   HundredGigE]</b> <i>interface-path-id</i> <b>Example:</b> <pre>RP/0/RP0/CPU0:router(config)# interface TenGigE 0/1/0/0</pre>	Enters interface configuration mode and specifies the Ethernet interface name and notation <i>rack/slot/module/port</i> . Possible interface types for this procedure are: <ul style="list-style-type: none"> <li>• GigabitEthernet</li> <li>• HundredGigE</li> <li>• TenGigE</li> </ul> <b>Note</b> <ul style="list-style-type: none"> <li>• The example indicates an 8-port 10-Gigabit Ethernet interface in modular services card slot 1.</li> </ul>
<b>Step 5</b>	<b>ipv4 address ip-address mask</b> <b>Example:</b> <pre>RP/0/RP0/CPU0:router(config-if)# ipv4 address 172.18.189.38 255.255.255.224</pre>	Assigns an IP address and subnet mask to the interface. <ul style="list-style-type: none"> <li>• Replace <i>ip-address</i> with the primary IPv4 address for the interface.</li> <li>• Replace <i>mask</i> with the mask for the associated IP subnet. The network mask can be specified in either of two ways:               <ul style="list-style-type: none"> <li>• The network mask can be a four-part dotted decimal address. For example, 255.0.0.0 indicates that each bit equal to 1 means that the corresponding address bit belongs to the network address.</li> <li>• The network mask can be indicated as a slash (/) and number. For example, /8 indicates that the first 8 bits of the mask are ones, and the corresponding bits of the address are network address.</li> </ul> </li> </ul>
<b>Step 6</b>	<b>flow-control {bidirectional  egress   ingress}</b> <b>Example:</b> <pre>RP/0/RP0/CPU0:router(config-if)# flow control ingress</pre>	(Optional) Enables the sending and processing of flow control pause frames. <ul style="list-style-type: none"> <li>• <b>egress</b>—Enables the sending of flow control pause frames in egress.</li> <li>• <b>ingress</b>—Enables the processing of received pause frames on ingress.</li> <li>• <b>bidirectional</b>—Enables the sending of flow control pause frames in egress and the processing of received pause frames on ingress.</li> </ul>
<b>Step 7</b>	<b>mtu bytes</b>	(Optional) Sets the MTU value for the interface.



	Command or Action	Purpose
	<p><b>Example:</b></p> <pre>RP/0/RP0/CPU0:router(config-if)# mtu 1448</pre>	<ul style="list-style-type: none"> <li>The default is 1514 bytes for normal frames and 1518 bytes for 802.1Q tagged frames.</li> <li>The range for Gigabit Ethernet and 10-Gigabit Ethernet mtu values is 64 bytes to 65535 bytes.</li> </ul>
<b>Step 8</b>	<p><b>mac-address</b> <i>value1.value2.value3</i></p> <p><b>Example:</b></p> <pre>RP/0/RP0/CPU0:router(config-if)# mac address 0001.2468.ABCD</pre>	<p>(Optional) Sets the MAC layer address of the Management Ethernet interface.</p> <ul style="list-style-type: none"> <li>The values are the high, middle, and low 2 bytes, respectively, of the MAC address in hexadecimal. The range of each 2-byte value is 0 to ffff.</li> </ul>
<b>Step 9</b>	<p><b>negotiation auto</b></p> <p><b>Example:</b></p> <pre>RP/0/0/CPU0:router(config-if)# negotiation auto</pre>	<p>(Optional) Enables autonegotiation on a Gigabit Ethernet interface.</p> <ul style="list-style-type: none"> <li>Autonegotiation must be explicitly enabled on both ends of the connection, or speed and duplex settings must be configured manually on both ends of the connection.</li> <li>If autonegotiation is enabled, any speed or duplex settings that you configure manually take precedence.</li> </ul> <p><b>Note</b></p> <ul style="list-style-type: none"> <li>The <b>negotiation auto</b> command is available on Gigabit Ethernet and Fast Ethernet interfaces only.</li> </ul>
<b>Step 10</b>	<p><b>no shutdown</b></p> <p><b>Example:</b></p> <pre>RP/0/RP0/CPU0:router(config-if)# no shutdown</pre>	Removes the shutdown configuration, which forces an interface administratively down.
<b>Step 11</b>	<p><b>end</b> or <b>commit</b></p> <p><b>Example:</b></p> <pre>RP/0/RP0/CPU0:router(config-if)# end</pre> <p>or</p> <pre>RP/0/RP0/CPU0:router(config-if)# commit</pre>	<p>Saves configuration changes.</p> <ul style="list-style-type: none"> <li>When you issue the <b>end</b> command, the system prompts you to commit changes: <pre>Uncommitted changes found, commit them before   exiting(yes/no/cancel)? [cancel]:</pre> </li> <li>Entering <b>yes</b> saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.</li> <li>Entering <b>no</b> exits the configuration session and returns the router to EXEC mode without committing the configuration changes.</li> </ul>

	Command or Action	Purpose
		<ul style="list-style-type: none"> <li>Entering <b>cancel</b> leaves the router in the current configuration session without exiting or committing the configuration changes.</li> <li>Use the <b>commit</b> command to save the configuration changes to the running configuration file and remain within the configuration session.</li> </ul>
<b>Step 12</b>	<b>show interfaces [GigabitEthernet   TenGigE] interface-path-id</b>  <b>Example:</b>  RP/0/RP0/CPU0:router# show interfaces TenGigE 0/3/0/0	(Optional) Displays statistics for interfaces on the router.

**What to do next**

To configure MAC Accounting on the Ethernet interface, see the “Configuring MAC Accounting on an Ethernet Interface” section later in this module.

To configure an AC on the Ethernet port for Layer 2 VPN implementation, see the “Configuring a L2VPN Ethernet Port” section later in this module.

To attach Layer 3 service policies, such as Multiprotocol Label Switching (MPLS) or Quality of Service (QoS), to the Ethernet interface, refer to the appropriate Cisco IOS XR software configuration guide.

**Configuring a Fast Ethernet Interface****What to do next**

- To configure an AC on the Fast Ethernet port for Layer 2 VPN implementation, see the “Configuring a L2VPN Ethernet Port” section later in this module.
- To attach Layer 3 service policies, such as Multiprotocol Label Switching (MPLS) or Quality of Service (QoS), to the Fast Ethernet interface, refer to the appropriate Cisco ASR 9000 Series Router or Cisco IOS XR software configuration guide.

**Configuring MAC Accounting on an Ethernet Interface**

This task explains how to configure MAC accounting on an Ethernet interface. MAC accounting has special show commands, which are illustrated in this procedure. Otherwise, the configuration is the same as configuring a basic Ethernet interface, and the steps can be combined in one configuration session. See “[Configuring Gigabit Ethernet Interfaces](#)” in this module for information about configuring the other common parameters for Ethernet interfaces.

**SUMMARY STEPS**

- configure**
- interface [GigabitEthernet | TenGigE | fastethernet] interface-path-id**
- ipv4 address ip-address mask**

4. `mac-accounting {egress | ingress}`
5. `end` or `commit`
6. `show mac-accounting type location instance`

## DETAILED STEPS

	Command or Action	Purpose
Step 1	<p><b>configure</b></p> <p><b>Example:</b></p> <pre>RP/0/RP0/CPU0:router# configure</pre>	Enters global configuration mode.
Step 2	<p><b>interface [GigabitEthernet   TenGigE   fastethernet]</b> <i>interface-path-id</i></p> <p><b>Example:</b></p> <pre>RP/0/RP0/CPU0:router(config)# interface TenGigE 0/1/0/0</pre>	<p>Physical interface or virtual interface.</p> <p><b>Note</b></p> <ul style="list-style-type: none"> <li>• Use the <b>show interfaces</b> command to see a list of all interfaces currently configured on the router.</li> </ul> <p>For more information about the syntax for the router, use the question mark (?) online help function.</p>
Step 3	<p><b>ipv4 address ip-address mask</b></p> <p><b>Example:</b></p> <pre>RP/0/RP0/CPU0:router(config-if)# ipv4 address 172.18.189.38 255.255.255.224</pre>	<p>Assigns an IP address and subnet mask to the interface.</p> <ul style="list-style-type: none"> <li>• Replace <i>ip-address</i> with the primary IPv4 address for the interface.</li> <li>• Replace <i>mask</i> with the mask for the associated IP subnet. The network mask can be specified in either of two ways: <ul style="list-style-type: none"> <li>• The network mask can be a four-part dotted decimal address. For example, 255.0.0.0 indicates that each bit equal to 1 means that the corresponding address bit belongs to the network address.</li> <li>• The network mask can be indicated as a slash (/) and number. For example, /8 indicates that the first 8 bits of the mask are ones, and the corresponding bits of the address are network address.</li> </ul> </li> </ul>
Step 4	<p><b>mac-accounting {egress   ingress}</b></p> <p><b>Example:</b></p> <pre>RP/0/RP0/CPU0:router(config-if)# mac-accounting egress</pre>	<p>Generates accounting information for IP traffic based on the source and destination MAC addresses on LAN interfaces.</p> <ul style="list-style-type: none"> <li>• To disable MAC accounting, use the <b>no</b> form of this command.</li> </ul>
Step 5	<p><b>end</b> or <b>commit</b></p> <p><b>Example:</b></p> <pre>RP/0/RP0/CPU0:router(config-if)# end</pre>	<p>Saves configuration changes.</p> <ul style="list-style-type: none"> <li>• When you issue the <b>end</b> command, the system prompts you to commit changes:</li> </ul>

	Command or Action	Purpose
	or  RP/0/RP0/CPU0:router(config-if)# commit	Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:  <ul style="list-style-type: none"> <li>• Entering <b>yes</b> saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.</li> <li>• Entering <b>no</b> exits the configuration session and returns the router to EXEC mode without committing the configuration changes.</li> <li>• Entering <b>cancel</b> leaves the router in the current configuration session without exiting or committing the configuration changes.</li> <li>• Use the <b>commit</b> command to save the configuration changes to the running configuration file and remain within the configuration session.</li> </ul>
<b>Step 6</b>	<b>show mac-accounting</b> <i>type location instance</i>  <b>Example:</b>  RP/0/RP0/CPU0:router# show mac-accounting TenGigE location 0/2/0/4	Displays MAC accounting statistics for an interface.

## Configuring a L2VPN Ethernet Port

Use the following procedure to configure an L2VPN Ethernet port.



**Note** The steps in this procedure configure the L2VPN Ethernet port to operate in port mode.

To configure a point-to-point pseudowire xconnect on an AC, see the Implementing MPLS Layer 2 VPNs module of the Cisco IOS XR L2VPN and Ethernet Services Configuration Guide for the Cisco CRS Router.

To attach Layer 2 service policies, such as quality of service (QoS), to the Ethernet interface, refer to the appropriate Cisco IOS XR software configuration guide.

### SUMMARY STEPS

1. **configure**
2. **interface** [**GigabitEthernet** | **TenGigE**] *interface-path-id*
3. **l2transport**
4. **l2protocol** **cpsv** {**tunnel** | **reverse-tunnel**}
5. **l2protocol** {**cdp** | **pvst** | **stp** | **vtp**} {**forward** | **tunnel**}[**experimental bits**][**drop**}
6. **end** or **commit**
7. **show interfaces** [**GigabitEthernet** | **TenGigE**] *interface-path-id*

## DETAILED STEPS

	Command or Action	Purpose
<b>Step 1</b>	<b>configure</b> <b>Example:</b> RP/0/RP0/CPU0:router# configure	Enters global configuration mode.
<b>Step 2</b>	<b>interface [GigabitEthernet   TenGigE] interface-path-id</b> <b>Example:</b> RP/0/RP0/CPU0:router(config)# interface TenGigE 0/1/0/0	Enters interface configuration mode and specifies the Ethernet interface name and notation <i>rack/slot/module/port</i> . Possible interface types for this procedure are: <ul style="list-style-type: none"> <li>• GigabitEthernet</li> <li>• TenGigE</li> </ul>
<b>Step 3</b>	<b>l2transport</b> <b>Example:</b> RP/0/RP0/CPU0:router(config-if)# l2transport	Enables Layer 2 transport mode on a port and enter Layer 2 transport configuration mode.
<b>Step 4</b>	<b>l2protocol cpsv {tunnel   reverse-tunnel}</b> <b>Example:</b> RP/0/RP0/CPU0:router(config-if-l2)# l2protocol cpsv tunnel	Configures Layer 2 protocol tunneling and protocol data unit (PDU) filtering on an Ethernet interface. <ul style="list-style-type: none"> <li>• <b>tunnel</b>—Specifies L2PT encapsulation on frames as they enter the interface, and de-encapsulation on frames as they exit they interface.</li> <li>• <b>reverse-tunnel</b>—Specifies L2PT encapsulation on frames as they exit the interface, and de-encapsulation on frames as they enter the interface.</li> </ul>
<b>Step 5</b>	<b>l2protocol {cdp   pvst   stp   vtp} {[forward   tunnel][experimental bits] drop}</b> <b>Example:</b> RP/0/RP0/CPU0:router(config-if-l2)# l2protocol stp tunnel	Configures Layer 2 protocol tunneling and protocol data unit (PDU) filtering on an Ethernet interface for one of the following protocols: CDP, PVST+, STP, VTP, where: <p>Possible protocols and options are:</p> <ul style="list-style-type: none"> <li>• <b>cdp</b>—Cisco Discovery Protocol (CDP) tunneling and data unit parameters.</li> <li>• <b>pvst</b>—Configures VLAN spanning tree protocol tunneling and data unit parameters.</li> <li>• <b>stp</b>—spanning tree protocol tunneling and data unit parameters.</li> <li>• <b>vtp</b>—VLAN trunk protocol tunneling and data unit parameters.</li> <li>• <b>tunnel</b>—(Optional) Tunnels the packets associated with the specified protocol.</li> </ul>

	Command or Action	Purpose
		<ul style="list-style-type: none"> <li>• <b>experimental bits</b>—(Optional) Modifies the MPLS experimental bits for the specified protocol.</li> <li>• <b>drop</b>—(Optional) Drop packets associated with the specified protocol.</li> </ul>
<b>Step 6</b>	<p><b>end</b> or <b>commit</b></p> <p><b>Example:</b></p> <pre>RP/0/RP0/CPU0:router(config-if-12)# end</pre> <p>or</p> <pre>RP/0/RP0/CPU0:router(config-if-12)# commit</pre>	<p>Saves configuration changes.</p> <ul style="list-style-type: none"> <li>• When you issue the <b>end</b> command, the system prompts you to commit changes: <pre>Uncommitted changes found, commit them before   exiting(yes/no/cancel)? [cancel]:</pre> </li> <li>• Entering <b>yes</b> saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.</li> <li>• Entering <b>no</b> exits the configuration session and returns the router to EXEC mode without committing the configuration changes.</li> <li>• Entering <b>cancel</b> leaves the router in the current configuration session without exiting or committing the configuration changes.</li> <li>• Use the <b>commit</b> command to save the configuration changes to the running configuration file and remain within the configuration session.</li> </ul>
<b>Step 7</b>	<p><b>show interfaces</b> [<b>GigabitEthernet</b>   <b>TenGigE</b>] <i>interface-path-id</i></p> <p><b>Example:</b></p> <pre>RP/0/RP0/CPU0:router# show interfaces TenGigE 0/3/0/0</pre>	(Optional) Displays statistics for interfaces on the router.

To configure a point-to-point pseudowire xconnect on an AC, refer to these documents:

- Cisco ASR 9000 Series Aggregation Services Router L2VPN and Ethernet Services Configuration Guide
- Cisco ASR 9000 Series Aggregation Services Router VPN and Ethernet Services Command Reference

## Configuring LLDP



**Note** LLDP is not supported on the FP-X line cards.

This section includes the following configuration topics for LLDP:

## LLDP Default Configuration

This table shows the values of the LLDP default configuration on the Cisco CRS Router. To change the default settings, use the LLDP global configuration and LLDP interface configuration commands.

LLDP Function	Default
LLDP global state	Disabled
LLDP holdtime (before discarding)	120 seconds
LLDP timer (packet update frequency)	30 seconds
LLDP reinitialization delay	2 seconds
LLDP TLV selection	All TLVs are enabled for sending and receiving.
LLDP interface state	Enabled for both transmit and receive operation when LLDP is globally enabled.

## Enabling LLDP Globally

To run LLDP on the router, you must enable it globally. When you enable LLDP globally, all interfaces that support LLDP are automatically enabled for both transmit and receive operations.

You can override this default operation at the interface to disable receive or transmit operations. For more information about how to selectively disable LLDP receive or transmit operations for an interface, see the [Disabling LLDP Receive and Transmit Operation for an Interface](#).

To enable LLDP globally, complete the following steps:

### SUMMARY STEPS

1. **configure**
2. **lldp**
3. **end** or **commit**

### DETAILED STEPS

	Command or Action	Purpose
Step 1	<b>configure</b> <b>Example:</b>  RP/0/RP0/CPU0:router# <b>configure</b>	Enters global configuration mode.
Step 2	<b>lldp</b> <b>Example:</b>  RP/0/RP0/CPU0:router(config)# <b>lldp</b>	Enables LLDP globally for both transmit and receive operation on the system.
Step 3	<b>end</b> or <b>commit</b> <b>Example:</b>	Saves configuration changes.

	Command or Action	Purpose
	<pre>RP/0/RP0/CPU0:router(config)# <b>end</b></pre> <p>or</p> <pre>RP/0/RP0/CPU0:router(config)# <b>commit</b></pre>	<ul style="list-style-type: none"> <li>When you issue the <b>end</b> command, the system prompts you to commit changes: <pre>Uncommitted changes found, commit them before   exiting(yes/no/cancel)? [cancel]:</pre>           Entering <b>yes</b> saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.           Entering <b>no</b> exits the configuration session and returns the router to EXEC mode without committing the configuration changes.           Entering <b>cancel</b> leaves the router in the current configuration session without exiting or committing the configuration changes.         </li> <li>Use the <b>commit</b> command to save the configuration changes to the running configuration file and remain within the configuration session.</li> </ul>

## Configuring Global LLDP Operational Characteristics

The [LLDP Default Configuration](#) describes the default operational characteristics for LLDP. When you enable LLDP globally on the router using the **lldp** command, these defaults are used for the protocol.

To modify the global LLDP operational characteristics such as the LLDP neighbor information holdtime, initialization delay, or packet rate, complete the following steps:

### SUMMARY STEPS

1. **configure**
2. **lldp holdtime** *seconds*
3. **lldp reinit** *seconds*
4. **lldp timer** *seconds*
5. **end** or **commit**

### DETAILED STEPS

	Command or Action	Purpose
<b>Step 1</b>	<pre><b>configure</b></pre> <p><b>Example:</b></p> <pre>RP/0/RP0/CPU0:router# <b>configure</b></pre>	Enters global configuration mode.



	Command or Action	Purpose
Step 2	<b>lldp holdtime</b> <i>seconds</i> <b>Example:</b> <pre>RP/0/RP0/CPU0:router(config)#lldp holdtime 60</pre>	(Optional) Specifies the length of time that information from an LLDP packet should be held by the receiving device before aging and removing it.
Step 3	<b>lldp reinit</b> <i>seconds</i> <b>Example:</b> <pre>RP/0/RP0/CPU0:router(config)# lldp reinit 4</pre>	(Optional) Specifies the length of time to delay initialization of LLDP on an interface.
Step 4	<b>lldp timer</b> <i>seconds</i> <b>Example:</b> <pre>RP/0/RP0/CPU0:router(config)#lldp reinit 60</pre>	(Optional) Specifies the LLDP packet rate.
Step 5	<b>end</b> or <b>commit</b> <b>Example:</b> <pre>RP/0/RP0/CPU0:router(config)# end</pre> <p>OR</p> <pre>RP/0/RP0/CPU0:router(config)# commit</pre>	<p>Saves configuration changes.</p> <ul style="list-style-type: none"> <li>When you issue the <b>end</b> command, the system prompts you to commit changes: <pre>Uncommitted changes found, commit them before   exiting(yes/no/cancel)? [cancel]:</pre> </li> <li>Entering <b>yes</b> saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.</li> <li>Entering <b>no</b> exits the configuration session and returns the router to EXEC mode without committing the configuration changes.</li> <li>Entering <b>cancel</b> leaves the router in the current configuration session without exiting or committing the configuration changes.</li> <li>Use the <b>commit</b> command to save the configuration changes to the running configuration file and remain within the configuration session.</li> </ul>

## Disabling Transmission of Optional LLDP TLVs

Certain TLVs are classified as mandatory in LLDP packets, such as the Chassis ID, Port ID, and Time to Live (TTL) TLVs. These TLVs must be present in every LLDP packet. You can suppress transmission of certain other optional TLVs in LLDP packets.

To disable transmission of optional LLDP TLVs, complete the following steps:

### SUMMARY STEPS

1. **configure**

2. `lldp tlv-select tlv-name disable`
3. `end` or `commit`

## DETAILED STEPS

	Command or Action	Purpose
<b>Step 1</b>	<b>configure</b> <b>Example:</b> RP/0/RP0/CPU0:router# <b>configure</b>	Enters global configuration mode.
<b>Step 2</b>	<b>lldp tlv-select tlv-name disable</b> <b>Example:</b> RP/0/RP0/CPU0:router(config)# <b>lldp tlv-select system-capabilities disable</b>	(Optional) Specifies that transmission of the selected TLV in LLDP packets is disabled. The <i>tlv-name</i> can be one of the following LLDP TLV types: <ul style="list-style-type: none"> <li>• <b>management-address</b></li> <li>• <b>port-description</b></li> <li>• <b>system-capabilities</b></li> <li>• <b>system-description</b></li> <li>• <b>system-name</b></li> </ul>
<b>Step 3</b>	<b>end</b> or <b>commit</b> <b>Example:</b> RP/0/RP0/CPU0:router(config)# <b>end</b> or RP/0/RP0/CPU0:router(config)# <b>commit</b>	Saves configuration changes. <ul style="list-style-type: none"> <li>• When you issue the <b>end</b> command, the system prompts you to commit changes:   <pre>Uncommitted changes found, commit them before   exiting(yes/no/cancel)? [cancel]:</pre> </li> <li>• Entering <b>yes</b> saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.</li> <li>• Entering <b>no</b> exits the configuration session and returns the router to EXEC mode without committing the configuration changes.</li> <li>• Entering <b>cancel</b> leaves the router in the current configuration session without exiting or committing the configuration changes.</li> <li>• Use the <b>commit</b> command to save the configuration changes to the running configuration file and remain within the configuration session.</li> </ul>

## Disabling LLDP Receive and Transmit Operation for an Interface

When you enable LLDP globally on the router, all supported interfaces are automatically enabled for LLDP receive and transmit operation. You can override this default by disabling these operations for a particular interface.

To disable LLDP receive and transmit operations for an interface, complete the following steps:

### SUMMARY STEPS

1. **configure**
2. **interface GigabitEthernet 0/2/0/0**
3. **lldp**
4. **receive disable**
5. **transmit disable**
6. **end** or **commit**

### DETAILED STEPS

	Command or Action	Purpose
Step 1	<b>configure</b> <b>Example:</b>  RP/0/RP0/CPU0:router# <b>configure</b>	Enters global configuration mode.
Step 2	<b>interface GigabitEthernet 0/2/0/0</b> <b>Example:</b>  RP/0/RP0/CPU0:router(config)# <b>interface GigabitEthernet 0/2/0/0</b>	Enters interface configuration mode and specifies the Ethernet interface name and notation <i>rack/slot/module/port</i> . Possible interface types for this procedure are: <ul style="list-style-type: none"> <li>• GigabitEthernet</li> <li>• TenGigE</li> </ul>
Step 3	<b>lldp</b> <b>Example:</b>  RP/0/RP0/CPU0:router(config-if)# <b>lldp</b>	(Optional) Enters LLDP configuration mode for the specified interface.
Step 4	<b>receive disable</b> <b>Example:</b>  RP/0/RP0/CPU0:router(config-lldp)# <b>receive disable</b>	(Optional) Disables LLDP receive operations on the interface.
Step 5	<b>transmit disable</b> <b>Example:</b>  RP/0/RP0/CPU0:router(config-lldp)# <b>transmit disable</b>	(Optional) Disables LLDP transmit operations on the interface.
Step 6	<b>end</b> or <b>commit</b>	Saves configuration changes.

Command or Action	Purpose
<p><b>Example:</b></p> <pre>RP/0/RP0/CPU0:router(config)# <b>end</b></pre> <p>or</p> <pre>RP/0/RP0/CPU0:router(config)# <b>commit</b></pre>	<ul style="list-style-type: none"> <li>• When you issue the <b>end</b> command, the system prompts you to commit changes:  Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:</li> <li>• Entering <b>yes</b> saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.</li> <li>• Entering <b>no</b> exits the configuration session and returns the router to EXEC mode without committing the configuration changes.</li> <li>• Entering <b>cancel</b> leaves the router in the current configuration session without exiting or committing the configuration changes.</li> <li>• Use the <b>commit</b> command to save the configuration changes to the running configuration file and remain within the configuration session.</li> </ul>

## Verifying the LLDP Configuration

This section describes how you can verify the LLDP configuration both globally and for a particular interface.

### Verifying the LLDP Global Configuration

To verify the LLDP global configuration status and operational characteristics, use the **show lldp** command as shown in the following example:

```
RP/0/RSP0/CPU0:router# show lldp
Wed Apr 13 06:16:45.510 DST
Global LLDP information:
  Status: ACTIVE
  LLDP advertisements are sent every 30 seconds
  LLDP hold time advertised is 120 seconds
  LLDP interface reinitialisation delay is 2 seconds
```

If LLDP is not enabled globally, the following output appears when you run the **show lldp** command:

```
RP/0/RSP0/CPU0:router# show lldp
Wed Apr 13 06:42:48.221 DST
% LLDP is not enabled
```

### Verifying the LLDP Interface Configuration

To verify the LLDP interface status and configuration, use the **show lldp interface** command as shown in the following example:

```
RP/0/RSP0/CPU0:router# show lldp interface GigabitEthernet 0/1/0/7
Wed Apr 13 13:22:30.501 DST
```

```
GigabitEthernet0/1/0/7:
  Tx: enabled
  Rx: enabled
  Tx state: IDLE
  Rx state: WAIT FOR FRAME
```

To monitor and maintain LLDP on the system or get information about LLDP neighbors, use one of the following commands:

	Description
<b>clear lldp counters</b>	Resets LLDP traffic counters or LLDP neighbor information.
<b>show lldp entry</b>	Displays detailed information about LLDP neighbors.
<b>show lldp errors</b>	Displays LLDP error and overflow statistics.
<b>show lldp neighbors</b>	Displays information about LLDP neighbors.
<b>show lldp traffic</b>	Displays statistics for LLDP traffic.

## Creating Slices on a Router

This section describes the procedures to configure slices and port-groups on the Cisco ASR 9000 Series 5th Generation Line Cards, and includes the following topics:

### Overview

Slicing is a way to transform a shared network into a set of logical networks. A network slice is a logical group of components, such as Optics (QSFP/SFP), PHY, Network Processor, and Fabric Cards. The system already has the default slicing configured when you enable it. However, you can configure the slices as per your network or service requirement.

You can enable slice configuration on the following:

Routers:

- ASR 9902
- ASR 9903
  - A9903-8HG-PEC

Line cards:

- A9K-4HG-FLEX-SE/-TR
- A99-4HG-FLEX-SE/-TR

### Advantages and Benefits

- Each slice enables you to deliver traffic of up to 400 Gbps, which is the maximum capacity that an NPU can handle.

- Provides the capability to the users to choose a speed from 400GE to 10GE.

### Restrictions on Slice Configurations

- You must configure port speed on the slices in the same or decreasing order. Beginning from the highest port speed to the lowest port speed.

## Slice and Port Numbering

The following tables lists the possible group combinations on the line cards and routers:

**Table 4: Possible Configurations for Cisco ASR 9000 Series Routers and 5th Generation Line Cards**

Router/Line card	Release	Supported Configurations
ASR 9902	Release 7.4.1	Default: 1x100GE, 1x100GE, 10x10GE, 10x10GE  1x100GE, 1x100GE, 4x25GE, 10x10GE  1x100GE, 4x25GE, 4x25GE, 1x100GE  1x100GE, 1x100GE, 1x100GE, 1x100GE
	Release 7.5.1	Default: 1x100GE, 1x100GE, 10x10GE, 10x10GE  1x100GE, 1x100GE, 4x25GE, 10x10GE  1x100GE, 4x25GE, 4x25GE, 1x100GE  1x100GE, 1x100GE, 1x100GE, 1x100GE  <b>Note</b> The following configuration modes are not supported: <ul style="list-style-type: none"> <li>• 1x40GE_4x10GE</li> <li>• 5x1GE_5x10GE</li> </ul>
	Release 7.5.2	Added support for the following configuration mode(s): <ul style="list-style-type: none"> <li>• 5x1GE_5x10GE</li> </ul>

Router/Line card	Release	Supported Configurations
ASR 9903	Release 7.4.1	Default: 10x10GE, 10x10GE, 4x10GE, UNUSED  4x25GE, 4x25GE, 4x25GE, 4x25GE
	Release 7.5.1	Default: 10x10GE, 10x10GE, 4x10GE, UNUSED  4x25GE, 4x25GE, 4x25GE, 4x25GE
	Cisco ASR 9903 800G Multirate Port Expansion Card (A9903-8HG-PEC)	Release 7.4.1  Default: 4x25GE, 4x25GE, 4x25GE, 4x25GE  1x100GE, 1x100GE, 4x25GE, 10x10GE
A9K-4HG-FLEX-SE A9K-4HG-FLEX-TR A99-4HG-FLEX-SE A99-4HG-FLEX-TR		Default: 10x10GE, 10x10GE, 4x10GE, UNUSED  4x25GE, 4x25GE, 4x25GE, 4x25GE

## Configure Slices

All configurations can be accomplished by using appropriate values for the **hw-module** command.



**Note** Slice configuration is available on A9903-8HG-PEC. For the other ports on PECs, fixed ports, or slices, use the breakout commands.

To configure the slice, use the following command:

```
configure hw-module location location slice [slice_number ] config-mode
[1x100GE, 1x100GE, 4x25GE, 10x10GE]
```

Consider a scenario, where, you as a user or a customer wants to effectively utilize the bandwidth of the router. In this scenario, we are going to configure a slice on a ASR 9000 Series Router that has a A9903-8HG-PEC line card installed. With A9903-8HG-PEC line card, you can configure slice 4 and 5. The rest of the slices are reserved for other line cards and fixed boards.

To configure the slices on the router or the line card, perform the following steps:

1. Identify the router capability, the line cards installed on the router, and the maximum bandwidth supported. In this scenario, the maximum supported bandwidth is 400GE per Network Processor.
2. Configure slices on the line card or router by using the **configure hw-module location location slice [slice\_number ] config-mode** command.
  - a. Specify the slices, such as slice 4.

- b. Specify the speed on the ports, such as 4x25GE,4x25GE,4x25GE,4x25GE or 1x100GE,1x100GE,4x25GE,10x10GE.
- c. Save the configuration.

Now, you have successfully configured the slices on the router, which will allow you to use the network bandwidth as per your requirement, without letting the bandwidth getting wasted, which wasn't possible earlier with the port breakout functionality.

For example, you can use the 4x25GE bandwidth for a lower bandwidth consumption service, such as voice calling. And use the 1x100GE bandwidth for video calling service.

### Configuration Steps

```
router# configure terminal
router(config)# hw-module location 0/0/CPU0 slice 4 config-mode 4x25GE,4x25GE,4x25GE,4x25GE
/* Specify the slice where you configure the groups and specify the port speeds for each
group*/
router(config)# commit
```

### Verify Slice Configuration

Use this procedure to verify whether the slice is correctly configured.

#### Verification on ASR 9903

```
Router: #show controllers np valid-port-groups all location 0/0/CPU0
Tue Dec 21 10:47:37.522 UTC
```

```
Node: 0/0/CPU0:
```

```
-----
NP0 valid-port-groups is not supported
```

```
NP1 valid-port-groups is not supported
```

```
NP2 valid-port-groups is not supported
```

```
NP3 valid-port-groups is not supported
```

```
NP4: Valid Port Groups
```

SEL	Group 0	Group 1	Group 2	Group 3
DFLT	10x10GE	10x10GE	4x10GE	UNUSED
1	4x25GE	4x25GE	4x25GE	4x25GE

```
Notes:
```

```
DFLT: SFP10g port 0-23 (default)
```

```
DFLT is set by default or by clearing any non-default configuration using "no
hw-module loc <> slice config-mode"
```

```
1: SFP25g port 0-15
```

```
NP5: Valid Port Groups
```

SEL	Group 0	Group 1	Group 2	Group 3
DFLT	10x10GE	10x10GE	4x10GE	UNUSED
1	4x25GE	4x25GE	4x25GE	4x25GE



```
Notes:
  DFLT: SFP10g port 24-47 (default)
  DFLT is set by default or by clearing any non-default configuration using "no
hw-module loc <> slice config-mode"
  1: SFP25g port 32-47
```

NP6 valid-port-groups is not supported

NP7 valid-port-groups is not supported

NP8 valid-port-groups is not supported

Router: #

```
RP/0/RP0/CPU0:router#sh controllers np ports all location 0/0/CPU0
Tue Jan 11 10:43:35.581 UTC
Node: 0/0/CPU0:
```

```
-----
NP Bridge Fia                               Ports
-----
0  --    0  TenGigE0/0/0/0/0 - TenGigE0/0/0/0/3, TenGigE0/0/0/1/0 - TenGigE0/0/0/1/3,
HundredGigE0/0/0/2, FortyGigE0/0/0/3/0
1  --    1  FortyGigE0/0/0/4/0, HundredGigE0/0/0/5, TenGigE0/0/0/6/0 - TenGigE0/0/0/6/3,
FortyGigE0/0/0/7/0
2  --    2  FortyGigE0/0/0/8/0, FortyGigE0/0/0/9/0, TenGigE0/0/0/26 - TenGigE0/0/0/35,
FortyGigE0/0/0/11/0
3  --    3  FortyGigE0/0/0/12/0, HundredGigE0/0/0/13 - HundredGigE0/0/0/15
4  --    4  TwentyFiveGigE0/0/1/0 - TwentyFiveGigE0/0/1/15
5  --    5  TenGigE0/0/1/24 - TenGigE0/0/1/47
```

**Verification on A9K-4HG-FLEX-SE Line Cards**

```
RP/0/RSP0/CPU0:asr-sat-host1(config)#do show controllers np valid-port-groups all location
0/0/CPU0
Tue Dec 21 05:07:32.819 UTC
Node: 0/0/CPU0:
```

```
-----
NP0: Valid Port Groups
-----+-----+-----+-----+-----
SEL | Group 0          | Group 1          | Group 2          | Group 3
-----+-----+-----+-----+-----
DFLT 1x100GE      | 1x100GE          | 1x100GE          | 1x100GE

1  1x100GE/1x40GE/4x25GE | 1x100GE/1x40GE/4x25GE | 1x100GE/1x40GE/4x25GE | 1x100GE/1x40GE/4x25GE

2  1x100GE/1x40GE/4x25GE | 1x100GE/1x40GE/4x25GE | 1x100GE/1x40GE/4x25GE | 10x10GE

3  1x100GE/1x40GE/4x25GE | 1x100GE/1x40GE/4x25GE | 10x10GE              | 10x10GE

4  1x100GE/1x40GE/4x25GE | 10x10GE          | 10x10GE            | 10x10GE

5  10x10GE             | 10x10GE          | 10x10GE            | 10x10GE
```

```
Notes:
  DFLT : QSFP port 40, QSFP port 41, QSFP port 42, QSFP port 43 (default)
  Group 0: QSFP port 40 or SFP10g port 0-9 or SFP25g port 6-9
  Group 1: QSFP port 41 or SFP10g port 10-19 or SFP25g port 16-19
  Group 2: QSFP port 42 or SFP10g port 20-29 or SFP25g port 26-29
  Group 3: QSFP port 43 or SFP10g port 30-39 or SFP25g port 36-39
RP/0/RSP0/CPU0:asr-sat-host1(config)#
```

For information on breakout, see the [hw-module location](#) command.

# Configuration Examples for Ethernet

This section provides the following configuration examples:

## Configuring an Ethernet Interface: Example

The following example shows how to configure an interface for a 10-Gigabit Ethernet modular services card:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# interface TenGigE 0/0/0/1
RP/0/RP0/CPU0:router(config-if)# ipv4 address 172.18.189.38 255.255.255.224
RP/0/RP0/CPU0:router(config-if)# flow-control ingress
RP/0/RP0/CPU0:router(config-if)# mtu 1448
RP/0/RP0/CPU0:router(config-if)# mac-address 0001.2468.ABCD
RP/0/RP0/CPU0:router(config-if)# no shutdown
RP/0/RP0/CPU0:router(config-if)# end
Uncommitted changes found, commit them? [yes]: yes
```

```
RP/0/RP0/CPU0:router# show interfaces TenGigE 0/0/0/1

TenGigE0/0/0/1 is down, line protocol is down
  Hardware is TenGigE, address is 0001.2468.abcd (bia 0001.81a1.6b23)
  Internet address is 172.18.189.38/27
  MTU 1448 bytes, BW 10000000 Kbit
    reliability 0/255, txload Unknown, rxload Unknown
  Encapsulation ARPA,
  Full-duplex, 10000Mb/s, LR
  output flow control is on, input flow control is on
  loopback not set
  ARP type ARPA, ARP timeout 01:00:00
  Last clearing of "show interface" counters never
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
    0 packets input, 0 bytes, 0 total input drops
    0 drops for unrecognized upper-level protocol
  Received 0 broadcast packets, 0 multicast packets
    0 runts, 0 giants, 0 throttles, 0 parity
  0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
  0 packets output, 0 bytes, 0 total output drops
  Output 0 broadcast packets, 0 multicast packets
  0 output errors, 0 underruns, 0 applique, 0 resets
  0 output buffer failures, 0 output buffers swapped out
  0 carrier transitions
```

## Configuring a Fast Ethernet Interface: Example

The following example indicates how to configure an interface for a Fast Ethernet SPA:

```
RP/0/0/CPU0:router# configure
RP/0/0/CPU0:router(config)# interface fastethernet 0/0/2/0
RP/0/0/CPU0:router(config-if)# ipv4 address 172.30.1.2 255.255.255.224
RP/0/0/CPU0:router(config-if)# duplex full
```

```

RP/0/0/CPU0:router(config-if)# mtu 1514
RP/0/0/CPU0:router(config-if)# speed 100
RP/0/0/CPU0:router(config-if)# no shutdown
RP/0/0/CPU0:router(config-if)# end
Uncommitted changes found, commit them? [yes]: yes

RP/0/0/CPU0:router# show interfaces fastethernet 0/0/2/0

FastEthernet0/0/2/0 is up, line protocol is up

  Hardware is FastEthernet, address is 000f.f83b.30c8 (bia 000f.f83b.30c8)
  Internet address is 172.30.1.2/24
  MTU 1514 bytes, BW 1000000 Kbit
  reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA,
  Duplex unknown, 100Mb/s, TX, link type is force-up
  output flow control is off, input flow control is off
  loopback not set
  ARP type ARPA, ARP timeout 04:00:00
  Last clearing of "show interface" counters never
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
    0 packets input, 0 bytes, 0 total input drops
    0 drops for unrecognized upper-level protocol
  Received 0 broadcast packets, 0 multicast packets
    0 runts, 0 giants, 0 throttles, 0 parity
  0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
  0 packets output, 0 bytes, 0 total output drops
  Output 0 broadcast packets, 0 multicast packets
  0 output errors, 0 underruns, 0 applique, 0 resets
  0 output buffer failures, 0 output buffers swapped out
  0 carrier transitions

```

## Configuring MAC-Accounting: Example

This example indicates how to configure MAC-accounting on an Ethernet interface:

```

RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# interface TenGigE 0/0/0/2
RP/0/RP0/CPU0:router(config-if)# ipv4 address 172.18.189.38 255.255.255.224
RP/0/RP0/CPU0:router(config-if)# mac-accounting egress
RP/0/RP0/CPU0:router(config-if)# commit
RP/0/RP0/CPU0:router(config-if)# exit
RP/0/RP0/CPU0:router(config)# exit

```

## Configuring a Layer 2 VPN AC: Example

The following example indicates how to configure a Layer 2 VPN AC on an Ethernet interface:

```

RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# interface TenGigE 0/0/0/2
RP/0/RP0/CPU0:router(config-if)# l2transport

```

```
RP/0/RP0/CPU0:router(config-if-l2)# l2protocol pvstcpsv  
RP/0/RP0/CPU0:router(config-if-l2)# commit
```

## Configuring LLDP: Examples

The following example shows how to enable LLDP globally on the router and modify the default LLDP operational characteristics:

```
RP/0/RP0/CPU0:router# configure  
RP/0/RP0/CPU0:router(config)# lldp  
RP/0/RP0/CPU0:router(config)# lldp holdtime 60  
RP/0/RP0/CPU0:router(config)# lldp reinit 4  
RP/0/RP0/CPU0:router(config)# lldp timer 60  
RP/0/RP0/CPU0:router(config)# commit
```

The following example shows how to disable a specific Gigabit Ethernet interface for LLDP transmission:

```
RP/0/RP0/CPU0:router# configure  
RP/0/RP0/CPU0:router(config)# interface GigabitEthernet 0/2/0/0  
RP/0/RP0/CPU0:router(config-if)# lldp  
RP/0/RP0/CPU0:router(config-lldp)# transmit disable
```

### Where to Go Next

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

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

For information about IPv6 see the Implementing Access Lists and Prefix Lists on Cisco IOS XR Software module in the Cisco IOS XR IP Addresses and Services Configuration Guide.

## How to Configure Interfaces in Breakout Mode

### Information About Breakout

The router supports transmission of traffic in the breakout mode. The breakout mode enables a 100GbE or 400GbE port to be split into multiple independent and logical GbE ports.

Breakout Mode options:

- 1x400GbE
- 1x100GbE
- 2x100GbE
- 4x100GbE

- 2x40GbE
- 1x40GbE
- 4x25GbE
- 4x10GbE



**Note** The supported breakout mode is dependent on the port and optic transceiver module.

## Breakout Configuration: Examples

These examples show how to configure breakout in a port.

### Configure 4x25GbE Breakout in a Port

This example shows how to configure 4x25GbE breakout in a port:

```
RP/0/RP0/CPU0:Router# configure
RP/0/RP0/CPU0:Router(config)# hw-module location 0/0/CPU0 bay 0 port 11 breakout
4xTwentyFiveGigE
RP/0/RP0/CPU0:Router(config)# commit
RP/0/RP0/CPU0:Router(config)# end
RP/0/RP0/CPU0:Router#
```

#### Verify a Breakout Configuration

Verifying the 4x25GbE breakout configuration:

```
RP/0/RP0/CPU0:Router# show ipv4 interfaces brief | inc 0/11/
    TwentyFiveGigE0/0/0/11/0      198.127.6.1    Up           Up           vpn24
    TwentyFiveGigE0/0/0/11/1      198.127.7.1    Up           Up           vpn7
    TwentyFiveGigE0/0/0/11/2      198.127.4.1    Up           Up           vpn4
    TwentyFiveGigE0/0/0/11/3      198.127.9.1    Up           Up           vpn21
```

#### Remove the Breakout Configuration

Removing the 4x25GbE breakout configuration:

```
RP/0/RP0/CPU0:uut# configure
RP/0/RP0/CPU0:uut(config)# no hw-module location 0/0/CPU0 bay 0 port 11 breakout
4xTwentyFiveGigE
RP/0/RP0/CPU0:uut(config)# commit
RP/0/RP0/CPU0:uut(config)# end
```

### Configure 4x100GbE Breakout in a Port

This example shows how to configure 4x100GbE breakout in a port:

```
RP/0/RP0/CPU0:Router# configure
RP/0/RP0/CPU0:Router(config)# hw-module location 0/0/CPU0 port 0 breakout 4xHundredGigE
RP/0/RP0/CPU0:Router(config)# commit
RP/0/RP0/CPU0:Router(config)# end
RP/0/RP0/CPU0:Router#
```

### Verify a Breakout Configuration

Verifying the 4x100GbE breakout configuration:

```
RP/0/RP0/CPU0:Router# show ipv4 interfaces brief | inc 0/11/
HundredGigE0/0/0/11/0      198.127.6.1      Up              Up              vpn24
HundredGigE0/0/0/11/1      198.127.7.1      Up              Up              vpn7
HundredGigE0/0/0/11/2      198.127.4.1      Up              Up              vpn4
HundredGigE0/0/0/11/3      198.127.9.1      Up              Up              vpn21
```

### Remove the Breakout Configuration

Removing the 4x100GbE breakout configuration:

```
RP/0/RP0/CPU0:uut# configure
RP/0/RP0/CPU0:uut(config)# no hw-module location 0/0/CPU0 port 0 breakout 4xHundredGigE
RP/0/RP0/CPU0:uut(config)# commit
RP/0/RP0/CPU0:uut(config)# end
```