



Configuring Cisco Discovery Protocol

Cisco Discovery Protocol (CDP) is a media- and protocol-independent protocol that runs on all Cisco-manufactured equipment including routers, bridges, access and communication servers, and switches. Using CDP, you can view information about all the Cisco devices that are directly attached to the device.

- [Prerequisites for Implementing CDP, on page 1](#)
- [Information About Implementing CDP, on page 1](#)
- [Enabling CDP, on page 3](#)
- [Modifying CDP Default Settings, on page 3](#)
- [Monitoring CDP, on page 4](#)

Prerequisites for Implementing CDP

To enable CDP, you must install the CDP package on your router.

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.

Information About Implementing CDP

CDP is primarily used to obtain protocol addresses of neighboring devices and discover the platform of those devices. CDP can also be used to display information about the interfaces your router uses. CDP is media- and protocol-independent, and runs on all equipment manufactured by Cisco, including routers, bridges, access servers, and switches.

Use of SNMP with the CDP MIB allows network management applications to learn the device type and the SNMP agent address of neighboring devices and to send SNMP queries to those devices. CDP uses the CISCO-CDP-MIB.

CDP runs on all media that support Subnetwork Access Protocol (SNAP), including LAN, Frame Relay, and ATM physical media. CDP runs over the data link layer only. Therefore, two systems that support different network-layer protocols can learn about each other.

Each device configured for CDP sends periodic messages, known as *advertisements*, to a multicast address. Each device advertises at least one address at which it can receive SNMP messages. The advertisements also contain time-to-live, or hold-time, information, which indicates the length of time a receiving device holds

CDP information before discarding it. Each device also listens to the periodic CDP messages sent by others to learn about neighboring devices and determine when their interfaces to the media go up or down.

CDP Version-2 (CDPv2) is the most recent release of the protocol and provides more intelligent device tracking features. These features include a reporting mechanism that allows for more rapid error tracking, thereby reducing costly downtime. Reported error messages can be sent to the console or to a logging server, and can cover instances of unmatching native VLAN IDs (IEEE 802.1Q) on connecting ports, and unmatching port duplex states between connecting devices.

CDPv2 **show** commands can provide detailed output on VLAN Trunking Protocol (VTP) management domain and duplex modes of neighbor devices, CDP-related counters, and VLAN IDs of connecting ports.

Type-length-value fields (TLVs) are blocks of information embedded in CDP advertisements. This table summarizes the TLV definitions for CDP advertisements.

Table 1: Type-Length-Value Definitions for CDPv2

TLV	Definition
Device-ID TLV	Identifies the device name in the form of a character string.
Address TLV	Contains a list of network addresses of both receiving and sending devices.
Port-ID TLV	Identifies the port on which the CDP packet is sent.
Capabilities TLV	Describes the functional capability for the device in the form of a device type; for example, a switch.
Version TLV	Contains information about the software release version on which the device is running.
Platform TLV	Describes the hardware platform name of the device, for example, Cisco 4500.
VTP Management Domain TLV	Advertises the system's configured VTP management domain name-string. Used by network operators to verify VTP domain configuration in adjacent network nodes.
Native VLAN TLV	Indicates, per interface, the assumed VLAN for untagged packets on the interface. CDP learns the native VLAN for an interface. This feature is implemented only for interfaces that support the IEEE 802.1Q protocol.
Full/Half Duplex TLV	Indicates status (duplex configuration) of CDP broadcast interface. Used by network operators to diagnose connectivity problems between adjacent network elements.

How to Implement CDP on Cisco IOS XR Software

Enabling CDP

To enable CDP, you must first enable CDP globally on the router and then enable CDP on a per-interface basis. This example explains how to enable CDP globally on the router and then enable CDP on an interface.

```
Router:# configure
Router(config):# cdp
Router(config):# commit

Router:# configure
Router(config):# interface hundredGigE 0/0/0/4
Router(config-if):# cdp
Router(config-if):# commit
```

Modifying CDP Default Settings

This task explains how to modify the default version, hold-time setting, and timer settings.



Note The commands can be entered in any order.

Step 1 **configure**

Example:

```
RP/0/RP0/CPU0:router# configure
```

Enters mode.

Step 2 **cdp advertise v1**

Example:

```
RP/0/RP0/CPU0:router(config)# cdp advertise v1
```

Configures CDP to use only version 1 (CDPv1) in communicating with neighboring devices.

- By default, when CDP is enabled, the router sends CDPv2 packets. CDP also sends and receives CDPv1 packets if the device with which CDP is interacting does not process CDPv2 packets.
- In this example, the router is configured to send and receive only CDPv1 packets.

Step 3 **cdp holdtime *seconds***

Example:

```
RP/0/RP0/CPU0:router(config)# cdp holdtime 30
```

Specifies the amount of time that the receiving networking device will hold a CDP packet sent from the router before discarding it.

- By default, when CDP is enabled, the receiving networking device holds a CDP packet for 180 seconds before discarding it.

Note The CDP hold time must be set to a higher number of seconds than the time between CDP transmissions, which is set with the **cdp timer** command.

- In this example, the value of hold-time for the *seconds* argument is set to 30.

Step 4 **cdp timer** *seconds*

Example:

```
RP/0/RP0/CPU0:router(config)# cdp timer 20
```

Specifies the frequency at which CDP update packets are sent.

- By default, when CDP is enabled, CDP update packets are sent at a frequency of once every 60 seconds.

Note A lower timer setting causes CDP updates to be sent more frequently.

- In this example, CDP update packets are configured to be sent at a frequency of once every 20 seconds.

Step 5 Use the **commit** or **end** command.

commit —Saves the configuration changes and remains within the configuration session.

end —Prompts user to take one of these actions:

- **Yes** — Saves configuration changes and exits the configuration session.
- **No** —Exits the configuration session without committing the configuration changes.
- **Cancel** —Remains in the configuration session, without committing the configuration changes.

Step 6 (Optional) **show cdp**

Example:

```
RP/0/RP0/CPU0:router# show cdp
```

Displays global CDP information.

The output displays the CDP version running on the router, the hold time setting, and the timer setting.

Monitoring CDP

This task shows how to monitor CDP.



Note The commands can be entered in any order.

Step 1 `show cdp entry` *{* | entry-name}* [**protocol** | **version**]

Example:

```
RP/0/RP0/CPU0:router# show cdp entry *
```

Displays information about a specific neighboring device or all neighboring devices discovered using CDP.

Step 2 `show cdp interface` [*type interface-path-id* | **location node-id**]

Example:

```
RP/0/RP0/CPU0:router# show cdp interface pos 0/0/0/1
```

Displays information about the interfaces on which CDP is enabled.

Step 3 `show cdp neighbors` [*type interface-path-id* | **location node-id**] [**detail**]

Example:

```
RP/0/RP0/CPU0:router# show cdp neighbors
```

Displays detailed information about neighboring devices discovered using CDP.

Step 4 `show cdp traffic` [**location node-id**]

Example:

```
RP/0/RP0/CPU0:router# show cdp traffic
```

Displays information about the traffic gathered between devices using CDP.

Examples

The following is sample output for the `show cdp neighbors` command:

```
RP/0/RP0/CPU0:router# show cdp neighbors
```

```
Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater
```

Device ID	Local Intrfce	Holdtme	Capability	Platform	Port ID
NCS5500	Hu0/0/0/4	15	R	NCS-5500	Hu0/0/0/4

The following is sample output for the `show cdp neighbors` command. In this example, the optional *type instance* arguments are used in conjunction with the **detail** optional keyword to display detailed information about a CDP neighbor. The output includes information on both IPv4 and IPv6 addresses.

```
RP/0/RP0/CPU0:router# show cdp neighbors hundredGigE 0/0/0/4 detail
```

```
-----
Device ID: NCS5500
SysName : NCS5500
Entry address(es):
```

```

    IPv4 address: 40.0.0.2
    IPv6 address: 10:10:10:10::1
Platform: cisco NCS-5500, Capabilities: Router
Interface: HundredGigE0/0/0/4
Port ID (outgoing port): HundredGigE0/0/0/4
Holdtime : 13 sec

Version :
7.1.1.112I

advertisement version: 2
Duplex: full

```

The following is sample output for the **show cdp entry** command. In this example, the optional *entry* argument is used to display entry information related to a specific CDP neighbor.

```

RP/0/RP0/CPU0:router# show cdp entry NCS5500
-----
Device ID: NCS5500
SysName : NCS5500
Entry address(es):
    IPv4 address: 40.0.0.2
    IPv6 address: 10:10:10:10::1
Platform: cisco NCS-5500, Capabilities: Router
Interface: HundredGigE0/0/0/4
Port ID (outgoing port): HundredGigE0/0/0/4
Holdtime : 11 sec

Version :
7.1.1.112I

advertisement version: 2
Duplex: full

```

The following is sample output for the **show cdp interface** command. In this example, CDP information related to interface 0/0/0/4 is displayed.

```

RP/0/RP0/CPU0:router# show cdp interface hundredGigE 0/0/0/4

HundredGigE0/0/0/4 is Up
Encapsulation ether
Sending CDP packets every 60 seconds
Holdtime is 180 seconds

```

The following is sample output for the **show cdp traffic** command:

```

RP/0/RP0/CPU0:router# show cdp traffic

CDP counters :
  Packets output: 10, Input: 39
  Hdr syntax: 0, Chksum error: 0, Encaps failed: 0
  No memory: 0, Invalid packet: 0, Truncated: 0
  CDP version 1 advertisements output: 0, Input: 0
  CDP version 2 advertisements output: 10, Input: 39
  Unrecognize Hdr version: 0, File open failed: 0

```