



Port Settings for Speed, Duplex, and Auto-negotiation

Objective

Use this topic to learn about, configure, verify, and troubleshoot port settings for speed, duplex, and auto-negotiation.

- [Feature history: port settings for an interface, on page 1](#)
- [About port settings for speed, duplex, and auto-negotiation, on page 2](#)
- [How port-settings applies interface settings, on page 2](#)
- [Restrictions for port settings, on page 3](#)
- [Configure port settings, on page 4](#)
- [Verify port settings, on page 4](#)

Feature history: port settings for an interface

Objective

This reference provides information about the features and enhancements that have been introduced for physical interfaces across different software releases.

Port settings

Table 1: Port settings

Description	Release information	Feature name
Configures speed, duplex, and negotiation auto in a single command.	Cisco IOS XE Catalyst SD-WAN Release 26.2.1	Port settings for speed, duplex, and auto-negotiation

About port settings for speed, duplex, and auto-negotiation

The **port-settings** command is an interface configuration command that groups speed, duplex, and auto-negotiation settings in one request. This grouping lets the device validate and apply the requested values in the required order.

Port-settings command behavior and benefits

Beginning with Cisco IOS XE Catalyst SD-WAN Release 26.2.1, the **port-settings** command is available on supported routing platforms. It configures speed, duplex, and negotiation auto in a single command. Specify the parameters in this order: speed, duplex, and autoneg (negotiation auto). The legacy **speed**, **duplex**, and **negotiation auto** commands remain supported.

Before Cisco IOS XE Catalyst SD-WAN Release 26.2.1, separate speed and duplex commands could fail if an intermediate configuration was invalid. The required command order depended on the current and desired settings.

Benefits

- The **port-settings** command applies speed and duplex settings in the correct order, thus preventing configuration failures.

How port-settings applies interface settings

Before Cisco IOS XE Catalyst SD-WAN Release 26.2.1, speed and duplex commands executed independently. A valid final configuration could fail when an intermediate configuration was invalid.

Summary

The port-settings process includes the following actors and components:

- The user supplies one or more speed, duplex, and auto-negotiation values in a single interface configuration command.
- Cisco IOS XE validates the requested values and their dependencies.
- The device applies the requested values in the required order.

Grouping the values prevents failures caused by an invalid intermediate configuration.

For example:

- Current settings: 100 Mbps and half duplex. Desired settings: 1000 Mbps and full duplex. Configuring 1000 Mbps first can fail because half duplex is not valid at 1000 Mbps.
- Current settings: 1000 Mbps and full duplex. Desired settings: 100 Mbps and half duplex. Configuring half duplex first can fail because half duplex is not valid at 1000 Mbps.

Workflow

These stages describe how Cisco IOS XE processes a grouped **port-settings** request:

1. Beginning with Cisco IOS XE Catalyst SD-WAN Release 26.2.1, the **port-settings** command accepts all the desired settings together. Cisco IOS XE can then validate and apply them in the correct order.

1. Receive the requested interface settings.

- The device receives the specified speed, duplex, and auto-negotiation parameters as one request.
- Omitted parameters retain their existing configured values.

The request is available for validation as a complete set of desired values. The device has the complete set of requested values for the next stage.

2. Validate the requested values.

- Cisco IOS XE checks the requested values and the dependencies between speed, duplex, and auto-negotiation.

The device identifies whether the requested combination is supported before applying the changes. The supported combination is ready to be applied in the required order.

3. Apply the requested values in order.

- The device applies speed, duplex, and auto-negotiation in the required order, avoiding the intermediate-state failures that could occur when legacy commands were entered separately.

The grouped request is reflected in the YANG running configuration. The requested interface settings are applied as one grouped configuration.

Result

The device applies the supported port settings as one grouped request and avoids failures caused by an invalid intermediate configuration.

What's next

To configure the feature, see [Configure port settings](#).

Restrictions for port settings

Upgrade restriction

Before upgrading a device from an earlier release—for example, Release 17.18.x—to Cisco IOS XE Catalyst SD-WAN Release 26.2.1, review any CLI template that uses separate speed, duplex, and negotiation auto commands. Update the template to use the new **port-settings** command; otherwise, the upgrade workflow may encounter problems.

RPC configuration restriction

Do not configure deprecated speed, duplex, or auto-negotiation YANG nodes and the new **port-settings** nodes in the same RPC. Conflicting values in one RPC can produce unexpected results.

Configure port settings

Configure supported speed, duplex, and auto-negotiation values in one interface-level **port-settings** command.

Use the **port-settings** command to submit the desired interface settings as one request. Specify parameters in the order speed, duplex, and autoneg.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Enter interface configuration mode for the target interface.

Example:

```
Device(config)# interface GigabitEthernet0/1/7
```

Step 3 Configure the speed, duplex mode, and auto-negotiation setting in one **port-settings** command.

Example:

This example configures 100 Mbps, half duplex, and disables auto-negotiation.

```
Device(config-if)# port-settings speed 100 duplex half autoneg disable
```

Step 4 Exit configuration mode and save the configuration according to your operational procedures.

Example:

```
Device(config-if)# end
```

The device applies the supported values in the required order. Check the running configuration after every grouped request, especially when the device rejects a parameter.

What to do next

Verify the operational interface state. See [Verify port settings](#).

Verify port settings

Objective

Use the verification commands and example output to confirm that the interface operates with the intended supported settings.

Verify the conventional interface configuration

Display the conventional interface running configuration:

```
Device# show running-config interface GigabitEthernet0/1/7
interface GigabitEthernet0/1/7
speed 100
duplex half
no negotiation auto
```

The conventional running configuration displays the legacy command representation: 100 Mbps, half duplex, and auto-negotiation disabled. This output remains in legacy form even when the grouped **port-settings** command supplies the configuration.

Verify the YANG interface configuration

Display the YANG running configuration for the interface:

```
Device# show running-config yang interface GigabitEthernet0/1/7
interface GigabitEthernet0/1/7
port-settings speed 100 duplex half autoneg disable
```

The YANG running configuration displays the same settings as one grouped **port-settings** entry. The speed, duplex, and autoneg values agree with the conventional output.

Verify the operational interface state

Display the operational state for the GigabitEthernet interface example:

```
Device# show interfaces GigabitEthernet0/1/7
GigabitEthernet0/1/7 is up, line protocol is up
Hardware is C8140-ES-8, address is 4880.02ed.c4ae (bia 4880.02ed.c4ae)
MTU 1500 bytes, BW 100000 Kbit/sec, DLY 100 usec,
reliability 255/255, txload 1/255, rxload 1/255
Encapsulation ARPA, loopback not set
Keepalive not supported
Half Duplex, 100Mbps, link type is force-up, media type is RJ45
output flow-control is off, input flow-control is on
```

This output reports the actual operational values for the interface. The interface and its line protocol are up. The port operates at half duplex and 100 Mbps, uses a force-up link type, and uses RJ45 media. Compare these values with the intended settings and the connected peer.

Result

Successful verification confirms that the legacy and new model representations are consistent and that the physical interface operates with the intended supported settings.

