



# Managing Switch Stacks

---

- [Finding Feature Information, on page 1](#)
- [Prerequisites for Switch Stacks, on page 1](#)
- [Restrictions for Switch Stacks, on page 2](#)
- [Information About Switch Stacks, on page 2](#)
- [How to Configure a Switch Stack, on page 3](#)
- [Configuration Examples for Switch Stacks, on page 3](#)
- [Feature History and Information for Switch Stacks, on page 11](#)

## Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <https://cfng.cisco.com/>. An account on Cisco.com is not required.

## Prerequisites for Switch Stacks

- All the switches in the stack must be running the same license level as the active switch. For information about license levels, see the *System Management* section of this guide.
- All the switches in the switch stack must be running compatible software versions.

# Restrictions for Switch Stacks

## Information About Switch Stacks

### Switch Stack Overview

### Switch Stack Bridge ID and MAC Address

The MAC address of the `sw1` determines the stack MAC address.

When the stack initializes, the MAC address of the `sw1` determines the bridge ID that identifies the stack in the network.

If the `sw1` changes, the MAC address of the new `sw1` determines the new bridge ID and stack MAC address.

If the entire switch stack reloads, the switch stack uses the MAC address of the `sw1`.

### Persistent MAC Address on the Switch Stack



---

**Note** You can also configure stack MAC persistency so that the stack MAC address never changes to the new MAC address, by using the `stack-mac persistent timer 0` command.

---

### Upgrading a Switch Running Incompatible Software

The auto-upgrade and auto-advise features enable a switch with software packages that are incompatible with the switch stack to be upgraded to a compatible software version so that it can join the switch stack.

### Switch Stack Management Connectivity

You manage the switch stack and the stack member interfaces through the `sw1`. You can use the CLI, SNMP, and supported network management applications such as CiscoWorks. You cannot manage stack members on an individual basis.

# How to Configure a Switch Stack

## Monitoring the Device Stack

*Table 1: Commands for Displaying Stack Information*

Command	Description
<code>show module</code>	Displays summary information about the stack.
<code>show switch detail</code>	Displays detailed information about the stack.
<code>show switch neighbors</code>	Displays the stack neighbors.
<code>show switch stack-ports</code> [summary]	Displays port information for the stack. Use the <b>summary</b> keyword to display the stack cable length, the stack link status, and the loopback status.
<code>show redundancy</code>	Displays the redundant system and the current processor information. The redundant system information includes the system uptime, standby failures, switchover reason, hardware, configured and operating redundancy mode. The current processor information displayed includes the active location, the software state, the uptime in the current state and so on.
<code>show redundancy state</code>	Displays all the redundancy states of the active and standby devices.

## Configuration Examples for Switch Stacks

### Switch Stack Configuration Scenarios

Most of these switch stack configuration scenarios assume that at least two devices are connected through their StackWise ports.

*Table 2: Configuration Scenarios*

Scenario		Result
Active switch election specifically determined by existing active switches	Connect two powered-on switch stacks through the StackWise ports.	Only one of the two active switches becomes the new active switch.

Scenario		Result
Active switch election specifically determined by the stack member priority value	<ol style="list-style-type: none"> <li>1. Connect two switches through their StackWise ports.</li> <li>2. Use the <b>switch</b> <i>stack-member-number priority new-priority-number</i> command to set one stack member with a higher member priority value.</li> <li>3. Restart both member switches at the same time.</li> </ol>	The stack member with the higher priority value is elected active switch.
Active switch election specifically determined by the configuration file	<p>Assuming that both member switches have the same priority value:</p> <ol style="list-style-type: none"> <li>1. Make sure that one stack member has a default configuration and that the other stack member has a saved (nondefault) configuration file.</li> <li>2. Restart both member switches at the same time.</li> </ol>	The stack member with the saved configuration file is elected active switch.
Active switch election specifically determined by the MAC address	<p>Assuming that both member switches have the same priority value, configuration file, and license level, restart both member switches at the same time.</p>	The stack member with the lower MAC address is elected active switch .
Stack member number conflict	<p>Assuming that one stack member has a higher priority value than the other stack member:</p> <ol style="list-style-type: none"> <li>1. Ensure that both member switches have the same stack member number. If necessary, use the <b>switch</b> <i>current-stack-member-number renumber new-stack-member-number</i> command.</li> <li>2. Restart both member switches at the same time.</li> </ol>	The stack member with the higher priority value retains its stack member number. The other stack member has a new stack member number.
Add a stack member	<ol style="list-style-type: none"> <li>1. Power off the new switch.</li> <li>2. Through their StackWise ports, connect the new switch to a powered-on switch stack.</li> <li>3. Power on the new switch.</li> </ol>	The active switch is retained. The new switch is added to the switch stack.
Active switch failure	Remove (or power off) the active switch.	

Scenario		Result
Add member switches	<ol style="list-style-type: none"> <li>1. Through their StackWise ports, connect devices.</li> <li>2. Power on all devices.</li> </ol>	<p>Two devices become active switches. One active switch has member switches. The other active switch remains as a standalone device.</p> <p>Use the Mode button and port LEDs on the device to identify which devices are active switches and which devices belong to each active switch.</p>

## Enabling the Persistent MAC Address Feature: Example

This example shows how to configure the persistent MAC address feature for a 7-minute time delay and to verify the configuration:

```
(config)# stack-mac persistent timer 7
WARNING: The stack continues to use the base MAC of the old Master
WARNING: as the stack MAC after a master switchover until the MAC
WARNING: persistency timer expires. During this time the Network
WARNING: Administrators must make sure that the old stack-mac does
WARNING: not appear elsewhere in this network domain. If it does,
WARNING: user traffic may be blackholed.
(config)# end
# show switch
Switch/Stack Mac Address : 0016.4727.a900
Mac persistency wait time: 7 mins

Switch# Role Mac Address Priority Version State
-----
*1      0016.4727.a900 1 P2B Ready
```

## show switch stack-ports summary Command Output: Example

Only Port 1 on stack member 2 is disabled.

```
# show switch stack-ports summary
# / Stack Neighbor Cable Link Link Sync # In
Port# Port Status Length OK Active OK Changes Loopback
To LinkOK
-----
1/1 OK 3 50 cm Yes Yes Yes 1 No
1/2 Down None 3 m Yes No Yes 1 No
2/1 Down None 3 m Yes No Yes 1 No
2/2 OK 3 50 cm Yes Yes Yes 1 No
3/1 OK 2 50 cm Yes Yes Yes 1 No
3/2 OK 1 50 cm Yes Yes Yes 1 No
```

Table 3: show switch stack-ports summary Command Output

Field	Description
Switch#/Port#	Member number and its stack port number.
Stack Port Status	Status of the stack port. <ul style="list-style-type: none"> <li>• Absent—No cable is detected on the stack port.</li> <li>• Down—A cable is detected, but either no connected neighbor is up, or the stack port is disabled.</li> <li>• OK—A cable is detected, and the connected neighbor is up.</li> </ul>
Neighbor	Switch number of the active member at the other end of the stack cable.
Cable Length	Valid lengths are 50 cm, 1 m, or 3 m. If the switch cannot detect the cable length, the value is <i>no cable</i> . The cable might not be connected, or the link might be unreliable.
Link OK	Whether the stack cable is connected and functional. There may or may not be a neighbor connected on the other end. The <i>link partner</i> is a stack port on a neighbor switch. <ul style="list-style-type: none"> <li>• No—There is no stack cable connected to this port or the stack cable is not functional.</li> <li>• Yes—There is a functional stack cable connected to this port.</li> </ul>
Link Active	Whether a neighbor is connected on the other end of the stack cable. <ul style="list-style-type: none"> <li>• No—No neighbor is detected on the other end. The port cannot send traffic over this link.</li> <li>• Yes—A neighbor is detected on the other end. The port can send traffic over this link.</li> </ul>
Sync OK	Whether the link partner sends valid protocol messages to the stack port. <ul style="list-style-type: none"> <li>• No—The link partner does not send valid protocol messages to the stack port.</li> <li>• Yes—The link partner sends valid protocol messages to the port.</li> </ul>
# Changes to LinkOK	The relative stability of the link. If a large number of changes occur in a short period of time, link flapping can occur.
In Loopback	Whether a stack cable is attached to a stack port on the member. <ul style="list-style-type: none"> <li>• No—At least one stack port on the member has an attached stack cable.</li> <li>• Yes—None of the stack ports on the member has an attached stack cable.</li> </ul>

## Software Loopback: Examples

In a stack with three members, stack cables connect all the members:

```
# show switch stack-ports summary
#
Sw#/Port#  Port      Neighbor  Cable   Link  Link  Sync  #Changes  In
           Status                Length  OK    Active OK    To LinkOK Loopback
-----
1/1        OK         3         50 cm   Yes   Yes   Yes   1         No
1/2        OK         2         3 m     Yes   Yes   Yes   1         No
2/1        OK         1         3 m     Yes   Yes   Yes   1         No
2/2        OK         3         50 cm   Yes   Yes   Yes   1         No
3/1        OK         2         50 cm   Yes   Yes   Yes   1         No
3/2        OK         1         50 cm   Yes   Yes   Yes   1         No
```

If you disconnect the stack cable from Port 1 on Switch 1, these messages appear:

```
01:09:55: %STACKMGR-4-STACK_LINK_CHANGE: Stack Port 2 Switch 3 has changed to state DOWN
01:09:56: %STACKMGR-4-STACK_LINK_CHANGE: Stack Port 1 Switch 1 has changed to state DOWN
```

```
# show switch stack-ports summary
#
Sw#/Port#  Port      Neighbor  Cable   Link  Link  Sync  #Changes  In
           Status                Length  OK    Active OK    To LinkOK Loopback
-----
1/1        Absent    None      No cable No    No    No    1         No
1/2        OK         2         3 m     Yes   Yes   Yes   1         No
2/1        OK         1         3 m     Yes   Yes   Yes   1         No
2/2        OK         3         50 cm   Yes   Yes   Yes   1         No
3/1        OK         2         50 cm   Yes   Yes   Yes   1         No
3/2        Down      None      50 cm   No    No    No    1         No
```

If you disconnect the stack cable from Port 2 on Switch 1, the stack splits.

Switch 2 and Switch 3 are now in a two-member stack connected through stack cables:

```
# show sw stack-ports summary
#
Sw#/Port#  Port      Neighbor  Cable   Link  Link  Sync  #Changes  In
           Status                Length  OK    Active OK    To LinkOK Loopback
-----
2/1        Down      None      3 m     No    No    No    1         No
2/2        OK         3         50 cm   Yes   Yes   Yes   1         No
3/1        OK         2         50 cm   Yes   Yes   Yes   1         No
3/2        Down      None      50 cm   No    No    No    1         No
```

Switch 1 is a standalone switch:

```
# show switch stack-ports summary
#
Sw#/Port#  Port      Neighbor  Cable   Link  Link  Sync  #Changes  In
           Status                Length  OK    Active OK    To LinkOK Loopback
-----
1/1        Absent    None      No cable No    No    No    1         Yes
1/2        Absent    None      No cable No    No    No    1         Yes
```

## Software Loopback with Connected Stack Cables: Examples

- On Port 1 on Switch 1, the port status is *Down*, and a cable is connected.

On Port 2 on Switch 1, the port status is *Absent*, and no cable is connected.

```
# show switch stack-ports summary
#
Sw#/Port#  Port      Neighbor  Cable   Link  Link  Sync  #Changes  In
           Status                Length  OK    Active OK    To LinkOK Loopback
-----
1/1        Down      None      50 Cm   No    No    No    1          No
1/2        Absent    None      No cable No    No    No    1          No
```

- In a *physical loopback*, a cable connects both stack ports on a switch. You can use this configuration to test

- Cables on a switch that is running properly
- Stack ports with a cable that works properly

```
# show switch stack-ports summary
#
Sw#/Port#  Port      Neighbor  Cable   Link  Link  Sync  #Changes  In
           Status                Length  OK    Active OK    To LinkOK Loopback
-----
2/1         OK        2         50 cm   Yes   Yes   Yes   1          No
2/2         OK        2         50 cm   Yes   Yes   Yes   1          No
```

The port status shows that

- Switch 2 is a standalone switch.
- The ports can send and receive traffic.

## Software Loopback with no Connected Stack Cable: Example

```
# show switch stack-ports summary
#
Sw#/Port#  Port      Neighbor  Cable   Link  Link  Sync  #Changes  In
           Status                Length  OK    Active OK    To LinkOK Loopback
-----
1/1        Absent    None      No cable No    No    No    1          Yes
1/2        Absent    None      No cable No    No    No    1          Yes
```

## Finding a Disconnected Stack Cable: Example

Stack cables connect all stack members. Port 2 on Switch 1 connects to Port 1 on Switch 2.

This is the port status for the members:

```
# show switch stack-ports summary
#
Sw#/Port#  Port      Neighbor  Cable   Link  Link  Sync  #Changes  In
           Status                Length  OK    Active OK    To LinkOK Loopback
-----
```



1/1	OK	2	50 cm	Yes	Yes	Yes	0	No
1/2	OK	2	50 cm	Yes	Yes	Yes	0	No
2/1	OK	1	50 cm	Yes	Yes	Yes	0	No
2/2	OK	1	50 cm	Yes	Yes	Yes	0	No

If you disconnect the cable from Port 2 on Switch 1, these messages appear:

```
%STACKMGR-4-STACK_LINK_CHANGE: Stack Port 1 Switch 2 has changed to state DOWN
```

```
%STACKMGR-4-STACK_LINK_CHANGE: Stack Port 2 Switch 1 has changed to state DOWN
```

This is now the port status:

```
# show switch stack-ports summary
#
Sw#/Port#  Port      Neighbor  Cable   Link  Link  Sync  #Changes  In
           Status                    Length  OK    Active OK    To LinkOK Loopback
-----
1/1        OK        2         50 cm   Yes   Yes   Yes   1         No
1/2        Absent    None      No cable No    No    No    2         No
2/1        Down      None      50 cm   No    No    No    2         No
2/2        OK        1         50 cm   Yes   Yes   Yes   1         No
```

Only one end of the cable connects to a stack port, Port 1 on Switch 2.

- The *Stack Port Status* value for Port 2 on Switch 1 is *Absent*, and the value for Port 1 on Switch 2 is *Down*.
- The *Cable Length* value is *No cable*.

Diagnosing the problem:

- Verify the cable connection for Port 2 on Switch 1.
- Port 2 on Switch 1 has a port or cable problem if
  - The *In Loopback* value is *Yes*.

or

- The *Link OK*, *Link Active*, or *Sync OK* value is *No*.

## Fixing a Bad Connection Between Stack Ports: Example

Stack cables connect all members. Port 2 on Switch 1 connects to Port 1 on Switch 2.

This is the port status:

```
# show switch stack-ports summary
#
Sw#/Port#  Port      Neighbor  Cable   Link  Link  Sync  #Changes  In
           Status                    Length  OK    Active OK    To LinkOK Loopback
-----
1/1        OK        2         50 cm   Yes   Yes   Yes   1         No
1/2        Down      None      50 cm   No    No    No    2         No
2/1        Down      None      50 cm   No    No    No    2         No
```

2/2      OK      1      50 cm      Yes      Yes      Yes      1      No

Diagnosing the problem:

- The Stack Port Status value is *Down*.
- Link OK, Link Active, and Sync OK values are *No*.
- The Cable Length value is *50 cm*. The switch detects and correctly identifies the cable.

The connection between Port 2 on Switch 1 and Port 1 on Switch 2 is unreliable on at least one of the connector pins.

## Additional References for Switch Stacks

### Related Documents

Related Topic	Document Title
Cabling and powering on a switch stack.	

### Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	<a href="https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi">https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi</a>

### Standards and RFCs

Standard/RFC	Title
None	—

### MIBs

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

**Technical Assistance**

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

## Feature History and Information for Switch Stacks

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use the Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to [www.cisco.com/go/cfn](http://www.cisco.com/go/cfn). An account on Cisco.com is not required.

**Table 4: Feature Information for Switch Stacks**

Release	Feature	Feature Information
	Switch Stack	A switch stack can have up to eight stacking-capable switches connected through their StackWise ports. The stack members work together as a unified system. Layer 2 and Layer 3 protocols present the entire switch stack as a single entity to the network.

