



Managing Switch Stacks

- [Prerequisites for Switch Stacks, on page 1](#)
- [Restrictions for Switch Stacks, on page 1](#)
- [Information About Switch Stacks, on page 2](#)
- [How to Configure a Switch Stack, on page 15](#)
- [Troubleshooting the Switch Stack, on page 21](#)
- [Monitoring the Device Stack, on page 22](#)
- [Configuration Examples for Switch Stacks, on page 23](#)
- [Additional References for Switch Stacks, on page 26](#)

Prerequisites for Switch Stacks

All stack members must run the same Cisco IOS software image to ensure compatibility among stack members. For switch stack hardware considerations, see the *Catalyst 2960-XR Switch Hardware Installation Guide*.

Restrictions for Switch Stacks

The following are restrictions for your switch stack configuration:

- The switch does not support mixed stacking. Stacking is supported only with other Catalyst 2960-XR switches.
- Auto-upgrade of stack can not be done when one of the switches in stack is with version Cisco IOS 15.2(3)E. This means that whenever any of the switches in the stack goes into a version mismatch, and if either the active stack is running Cisco IOS 15.2(3)E, or if a member is running Cisco 15.2(3)E, the member can not be auto-upgraded to the required version.

Information About Switch Stacks

Switch Stack Overview

A switch stack is a set of up to eight stacking-capable switches connected through their stack ports. You can connect only one switch type in a stack, or you can connect a mix of Catalyst 2960-XR, Catalyst 2960-X and Catalyst 2960-S switches in the stack. The stack can have one of these configurations:

- Homogeneous stack—A Catalyst 2960-XR only stack with only Catalyst 2960-XR switches as stack members. A homogenous stack can have up to 8 stack members.
- Mixed stack—A stack with a mix of Catalyst 2960-XR, Catalyst 2960-X and Catalyst 2960-S switches as stack members. A mixed stack of Catalyst 2960-XR and Catalyst 2960-X switches can have up to 8 stack members. A mixed stack that includes a Catalyst 2960-S switch can have up to 4 stack members.

The active stack controls the operation of the switch stack, and is the single point of stack-wide management. From the active stack, you configure:

- System-level (global) features that apply to all stack members
- Interface-level features for each stack member

The active stack contains the saved and running configuration files for the switch stack. The configuration files include the system-level settings for the switch stack and the interface-level settings for each stack member. Each stack member has a current copy of these files for back-up purposes.

Supported Features in a Switch Stack

The system-level features supported on the active switch are supported on the entire switch stack.

Encryption Features

If the active switch is running the cryptographic universal software image (supports encryption), the encryption features are available on the switch stack.

FlexStack-Plus

The stack members use the Cisco FlexStack-Plus technology to work together as a unified system. Layer 2 protocols support the entire switch stack as a single entity in the network.



Note Switch stacks running the LAN Base image do not support Layer 3 features.

The FlexStack-Plus bandwidth for a single stack port is 20 Gbps. With FlexStack-Plus technology, up to eight members can be joined into a single stack. In a mixed stack of Catalyst 2960-X and Catalyst 2960-S switches, FlexStack-Plus reverts to FlexStack capabilities of 10 Gbps stack port bandwidth and a maximum of four members per stack.

Switch Stack Membership

A switch stack has up to eight stack members connected through their stack ports. A switch stack always has one active switch.

A standalone device is a device stack with one stack member that also operates as the active switch. You can connect one standalone device to another to create a stack containing two stack members, with one of them as the active switch. You can connect standalone devices to an existing device stack to increase the stack membership.

Figure 1: Creating a Switch Stack from Two Standalone Switches

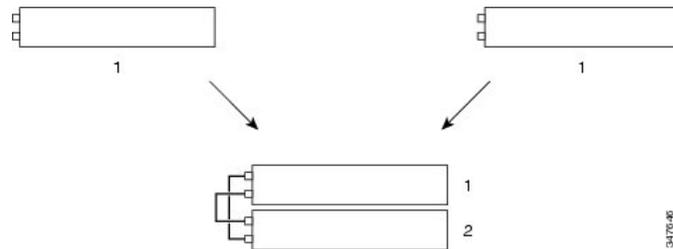
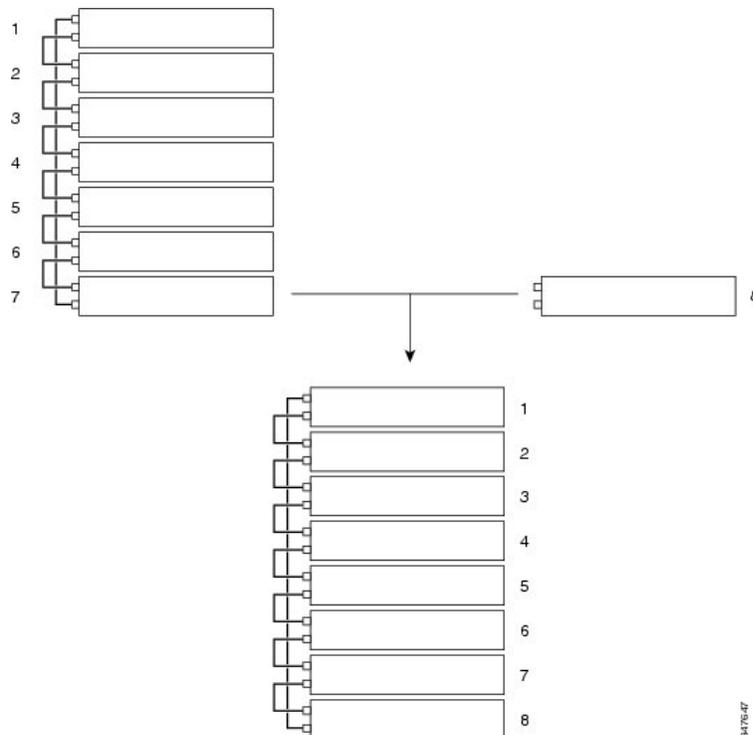


Figure 2: Adding a Standalone Switch to a Switch Stack



Changes to Switch Stack Membership

If you replace a stack member with an identical model, the new switch functions with exactly the same configuration as the replaced switch, assuming that the new switch (referred to as the provisioned switch) is using the same member number as the replaced switch.

The operation of the switch stack continues uninterrupted during membership changes unless you remove the active switch or you add powered-on standalone switches or switch stacks.

- Adding powered-on switches (merging) causes all switches to reload and elect a new active switch from among themselves. The newly elected active switch retains its role and configuration. All other switches retain their stack member numbers and use the stack configuration of the newly elected active switch.



Note In Cisco IOS XE 3.6.4E and later versions, when a new switch is powered-on as a standalone switch before it is added as part of the switch stack, only this switch is reloaded and not the whole switch stack.

Adding powered-on switches (merging) causes the active stack of the merging switch stacks to elect an active stack from among themselves. The reelected active stack retains its role and configuration as do its stack members. All remaining switches, including the former active stacks, reload and join the switch stack as stack members. They change their stack member numbers to the lowest available numbers and use the stack configuration of the reelected active stack.

- Removing powered-on stack members causes the switch stack to divide (partition) into two or more switch stacks, each with the same configuration. This can cause:
 - An IP address conflict in your network. If you want the switch stacks to remain separate, change the IP address or addresses of the newly created switch stacks.
 - A MAC address conflict between two members in the stack. You can use the **stack-mac update force** command to resolve the conflict.

If a newly created switch stack does not have an active switch or standby switch, the switch stack will reload and elect a new active switch.



Note Make sure that you power off the switches that you add to or remove from the switch stack.

After adding or removing stack members, make sure that the switch stack is operating at full bandwidth (40 Gbps). Press the Mode button on a stack member until the Stack mode LED is on. The last two right port LEDs on all switches in the stack should be green. Depending on the switch model, the last two right ports are 10-Gigabit Ethernet ports or small form-factor pluggable (SFP) module ports (10/100/1000 ports). If one or both of these LEDs are not green on any of the switches, the stack is not operating at full bandwidth.

It may take upto 4 seconds for stack convergence when a new stack member is added to the existing switch stack.

If you remove powered-on members but do not want to partition the stack:

- Power off the switches in the newly created switch stacks.
- Reconnect them to the original switch stack through their stack ports.
- Power on the switches.

For cabling and power considerations that affect switch stacks, see the *Catalyst 2960-XR Switch Hardware Installation Guide*.

Stack Member Numbers

The stack member number (1 to 8) identifies each member in the switch stack. The member number also determines the interface-level configuration that a stack member uses. You can display the stack member number by using the **show switch EXEC** command.

A new, out-of-the-box device (one that has not joined a device stack or has not been manually assigned a stack member number) ships with a default stack member number of 1. When it joins a device stack, its default stack member number changes to the lowest available member number in the stack.

Stack members in the same stack cannot have the same stack member number. Every stack member, including a standalone device, retains its member number until you manually change the number or unless the number is already being used by another member in the stack.

- If you manually change the stack member number by using the **switch current-stack-member-number renumber new-stack-member-number** command, the new number goes into effect after that stack member resets (or after you use the **reload slot stack-member-number** privileged EXEC command) and only if that number is not already assigned to any other members in the stack. Another way to change the stack member number is by changing the device_NUMBER environment variable.

If the number is being used by another member in the stack, the device selects the lowest available number in the stack.

If you manually change the number of a stack member and no interface-level configuration is associated with that new member number, that stack member resets to its default configuration.

You cannot use the **switch current-stack-member-number renumber new-stack-member-number** command on a provisioned device. If you do, the command is rejected.

- If you move a stack member to a different device stack, the stack member retains its number only if the number is not being used by another member in the stack. If it is being used, the device selects the lowest available number in the stack.
- If you merge device stacks, the device that join the device stack of a new active device select the lowest available numbers in the stack.

As described in the hardware installation guide, you can use the device port LEDs in Stack mode to visually determine the stack member number of each stack member.

In the **default** mode Stack LED will blink in green color only on the active switch. However, when we scroll the Mode button to **Stack** option - Stack LED will glow green on all the stack members.

When mode button is scrolled to **Stack** option, the switch number of each stack member will be displayed as LEDs on the first five ports of that switch. The switch number is displayed in binary format for all stack members. On the switch, the amber LED indicates value 0 and green LED indicates value 1.

Example for switch number 5 (Binary - 00101):

First five LEDs glow as follows on stack member with switch number 5.

- Port-1 : Amber
- Port-2 : Amber
- Port-3 : Green
- Port-4 : Amber
- Port-5 : Green

Similarly, the first five LEDs glow amber or green, depending on the switch number on all stack members.



- Note**
- Stack port will not go down but only transmission/reception will be disabled. The log message shown below will be displayed on the console. Once the peer end network port is converted to stack port, transmission/reception on this stack port will be enabled.

```
%STACKMGR-4-HSTACK_LINK_CONFIG: Verify peer stack port setting for hstack
StackPort-1 switch 5 (hostname-switchnumber)
```

Stack Member Priority Values

A higher priority value for a stack member increases the probability of it being elected active switch and retaining its stack member number. The priority value can be 1 to 15. The default priority value is 1. You can display the stack member priority value by using the **show switch EXEC** command.



- Note** We recommend assigning the highest priority value to the device that you prefer to be the active device. This ensures that the device is reelected as the active device if a reelection occurs.

To change the priority value for a stack member, use the **switch stack-member-number priority new priority-value** command. For more information, see the “Setting the Stack Member Priority Value” section.

The new priority value takes effect immediately but does not affect the current active device. The new priority value helps determine which stack member is elected as the new active device when the current active device or the device stack resets.

Switch Stack Bridge ID and MAC Address

The MAC address of the active switch determines the stack MAC address.

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

If the active switch changes, the MAC address of the new active switch determines the new bridge ID and stack MAC address.

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

Persistent MAC Address on the Switch Stack

You can use the persistent MAC address feature to set a time delay before the stack MAC address changes to the MAC address of the new active stack. When this feature is enabled, the stack MAC address changes in approximately 4 minutes. During this time, if the previous active stack rejoins the stack, the stack continues to use its MAC address as the stack MAC address, even if the switch is now a stack member and not the active stack. If the previous active stack does not rejoin the stack during this period, the switch stack takes the MAC address of the new active stack as the stack MAC address.

You can also configure stack MAC persistency so that the stack MAC address never changes to the new active switch MAC address.

Active and Standby Switch Election and Reelection

All stack members are eligible to be the active stack. If the active stack becomes unavailable, the remaining members elect a new active stack from among themselves.

The active switch is elected or reelected based on one of these factors and in the order listed:

1. The switch that is currently the active switch.
2. The switch with the highest stack member priority value.



Note We recommend assigning the highest priority value to the switch that you prefer to be the active switch. This ensures that the switch is reelected as active switch if a reelection occurs.

3. The switch that has the configuration file.
4. The switch with the lowest MAC address.

An active stack retains its role unless one of these events occurs:

- The switch stack is reset.*
- The active stack is removed from the switch stack.
- The active stack is reset or powered off.
- The active stack fails.
- The switch stack membership is increased by adding powered-on standalone switches or switch stacks.*

In the events marked by an asterisk (*), the current active stack *might* be reelected based on the listed factors.

When you power on or reset an entire switch stack, some stack members *might not* participate in the active stack election. Stack members that are powered on within the same 20-second time frame participate in the active stack election and have a chance to become the active stack. Stack members that are powered on after the 20-second time frame do not participate in this initial election and become stack members. All stack members participate in reelections. For all powering considerations that affect active-stack elections, see the “Switch Installation” chapter in the hardware installation guide.

The new active stack becomes available after a few seconds. In the meantime, the switch stack uses the forwarding tables in memory to minimize network disruption. The physical interfaces on the other available stack members are not affected during a new active stack election and reset.

After a new active stack is elected and the previous active stack becomes available, the previous active stack *does not* resume its role as the active stack.

For all powering considerations that affect active-stack elections, see the *Catalyst 2960-XR Switch Hardware Installation Guide*.

Switch Stack Configuration Files

The active switch has the saved and running configuration files for the switch stack. All stack members periodically receive synchronized copies of the configuration files from the active switch. If the active switch becomes unavailable, any stack member assuming the role of active switch has the latest configuration files.

The configuration files record these settings:

- System-level (global) configuration settings such as IP, STP, VLAN, and SNMP settings that apply to all stack members
- Stack member interface-specific configuration settings that are specific for each stack member



Note The interface-specific settings of the active switch are saved if the active switch is replaced without saving the running configuration to the startup configuration.

A new, out-of-box device joining a switch stack uses the system-level settings of that switch stack. If a device is moved to a different switch stack before it is powered on, that device loses its saved configuration file and uses the system-level configuration of the new switch stack. If the device is powered on as a standalone device before it joins the new switch stack, the stack will reload. When the stack reloads, the new device may become the device, retain its configuration and overwrite the configuration files of the other stack members.

The interface-specific configuration of each stack member is associated with the stack member number. Stack members retain their numbers unless they are manually changed or they are already used by another member in the same switch stack. If the stack member number changes, the new number goes into effect after that stack member resets.

- If an interface-specific configuration does not exist for that member number, the stack member uses its default interface-specific configuration.
- If an interface-specific configuration exists for that member number, the stack member uses the interface-specific configuration associated with that member number.

If you replace a failed member with an identical model, the replacement member automatically uses the same interface-specific configuration as the failed device. You do not need to reconfigure the interface settings. The replacement device (referred to as the provisioned device) must have the same stack member number as the failed device.

You back up and restore the stack configuration in the same way as you would for a standalone device configuration.

Offline Configuration to Provision a Stack Member

You can use the offline configuration feature to *provision* (to supply a configuration to) a new switch before it joins the switch stack. You can configure the stack member number, the switch type, and the interfaces associated with a switch that is not currently part of the stack. The configuration that you create on the switch stack is called the *provisioned configuration*. The switch that is added to the switch stack and that receives this configuration is called the *provisioned switch*.

You manually create the provisioned configuration through the **switch stack-member-number provision type** global configuration command. You must change the *stack-member-number* on the provisioned switch before you add it to the stack, and it must match the stack member number that you created for the new switch on the switch stack. The switch type in the provisioned configuration must match the switch type of the newly added switch. The provisioned configuration is automatically created when a switch is added to a switch stack and when no provisioned configuration exists.

When you configure the interfaces associated with a provisioned switch, the switch stack accepts the configuration, and the information appears in the running configuration. However, as the switch is not active,

any configuration on the interface is not operational and the interface associated with the provisioned switch does not appear in the display of the specific feature. For example, VLAN configuration information associated with a provisioned switch does not appear in the **show vlan** user EXEC command output on the switch stack.

The switch stack retains the provisioned configuration in the running configuration whether or not the provisioned switch is part of the stack. You can save the provisioned configuration to the startup configuration file by entering the **copy running-config startup-config** privileged EXEC command. The startup configuration file ensures that the switch stack can reload and can use the saved information whether or not the provisioned switch is part of the switch stack.

Effects of Adding a Provisioned Switch to a Switch Stack

When you add a provisioned Device to the switch stack, the stack applies either the provisioned configuration or the default configuration. This table lists the events that occur when the switch stack compares the provisioned configuration with the provisioned switch.

Table 1: Results of Comparing the Provisioned Configuration with the Provisioned Switch

Scenario		Result
The stack member numbers and the Device types match.	<ol style="list-style-type: none"> 1. If the stack member number of the provisioned switch matches the stack member number in the provisioned configuration on the stack, and 2. If the Device type of the provisioned switch matches the Device type in the provisioned configuration on the stack. 	The switch stack applies the provisioned configuration to the provisioned switch and adds it to the stack.
The stack member numbers match but the Device types do not match.	<ol style="list-style-type: none"> 1. If the stack member number of the provisioned switch matches the stack member number in the provisioned configuration on the stack, but 2. The Device type of the provisioned switch does not match the Device type in the provisioned configuration on the stack. 	<p>The switch stack applies the default configuration to the provisioned switch and adds it to the stack.</p> <p>The provisioned configuration is changed to reflect the new information.</p>
The stack member number is not found in the provisioned configuration.		<p>The switch stack applies the default configuration to the provisioned switch and adds it to the stack.</p> <p>The provisioned configuration is changed to reflect the new information.</p>
The stack member number of the provisioned switch is not found in the provisioned configuration.		The switch stack applies the default configuration to the provisioned switch and adds it to the stack.

If you add a provisioned switch that is a different type than specified in the provisioned configuration to a powered-down switch stack and then apply power, the switch stack rejects the (now incorrect) **switch stack-member-number provision type** global configuration command in the startup configuration file. However, during stack initialization, the nondefault interface configuration information in the startup configuration file for the provisioned interfaces (potentially of the wrong type) is executed. Depending on the differences between the actual Device type and the previously provisioned switch type, some commands are rejected, and some commands are accepted.



Note If the switch stack does not contain a provisioned configuration for a new Device, the Device joins the stack with the default interface configuration. The switch stack then adds to its running configuration with a **switch stack-member-number provision type** global configuration command that matches the new Device. For configuration information, see the *Provisioning a New Member for a Switch Stack* section.

Effects of Replacing a Provisioned Switch in a Switch Stack

When a provisioned switch in a switch stack fails, it is removed from the stack, and is replaced with another Device, the stack applies either the provisioned configuration or the default configuration to it. The events that occur when the switch stack compares the provisioned configuration with the provisioned switch are the same as those when you add a provisioned switch to a stack.

Effects of Removing a Provisioned Switch from a Switch Stack

If you remove a provisioned switch from the switch stack, the configuration associated with the removed stack member remains in the running configuration as provisioned information. To completely remove the configuration, use the **no switch stack-member-number provision** global configuration command.

Stack Protocol Version

Each software image includes a *stack protocol version*. The stack protocol version has a *major* version number and a *minor* version number (for example 1.4, where 1 is the major version number and 4 is the minor version number). Both version numbers determine the level of compatibility among the stack members. You can display the stack protocol version by using the **show platform stack manager all** privileged EXEC command.

The switches with the same Cisco IOS software version have the same stack protocol version. Such switches are fully compatible, and all features function properly across the switch stack. A device with the same Cisco IOS software version as the active switch can immediately join the switch stack.

If an incompatibility exists, the fully functional stack members generate a system message that describes the cause of the incompatibility on the specific stack members. The active switch sends the message to all stack members.

For more information, see the *Major Version Number Incompatibility Among Switches* procedure and the *Minor Version Number Incompatibility Among Switches* procedure.

Major Stack Protocol Version Number Incompatibility Among Stack-Capable Switches

Device with different major Cisco IOS software versions usually have different stack protocol versions. Device with different major version numbers are incompatible and cannot exist in the same switch stack.

Minor Stack Protocol Version Number Incompatibility Among Stack-Capable Switches

Switches with the same major version number but with a different minor version number are considered partially compatible. When connected to a switch stack, a partially compatible switch enters version-mismatch (VM) mode and cannot join the stack as a fully functioning member. The software detects the mismatched software and tries to upgrade (or downgrade) the switch in VM mode with the switch stack image or with a tar file image from the switch stack flash memory. The software uses the automatic upgrade (auto-upgrade) and the automatic advise (auto-advise) features.

The port LEDs on switches in version-mismatch mode will also remain off. Pressing the Mode button does not change the LED mode.

Auto-Upgrade

The purpose of the auto-upgrade feature is to allow a switch to be upgraded to a compatible software image, so that the switch can join the switch stack.

When a new switch attempts to join a switch stack, each stack member performs compatibility checks with itself and the new switch. Each stack member sends the results of the compatibility checks to the active stack, which uses the results to determine whether the switch can join the switch stack. If the software on the new switch is incompatible with the switch stack, the new switch enters version-mismatch (VM) mode.

If the auto-upgrade feature is enabled on the existing switch stack, the active stack automatically upgrades the new switch with the same software image running on a compatible stack member. Auto-upgrade starts a few minutes after the mismatched software is detected before starting.

By default, auto-upgrade is enabled (the **boot auto-copy-sw** global configuration command is enabled). You can disable auto-upgrade by using the **no boot auto-copy-sw** global configuration command on the active stack. You can check the status of auto-upgrade by using the **show boot** privileged EXEC command and by checking the *Auto upgrade* line in the display.

Auto-upgrade includes an auto-copy process and an auto-extract process.

- Auto-copy automatically copies the software image running on any stack member to the new switch to automatically upgrade it. Auto-copy occurs if auto-upgrade is enabled, if there is enough flash memory in the new switch, and if the software image running on the switch stack is suitable for the new switch.



Note A switch in VM mode might not run all released software. For example, new switch hardware is not recognized in earlier versions of software.

- Automatic extraction (auto-extract) occurs when the auto-upgrade process cannot find the appropriate software in the stack to copy to the new switch. In that case, the auto-extract process searches all switches in the stack for the tar file needed to upgrade the switch stack or the new switch. The tar file can be in any flash file system in the switch stack or in the new switch. If a tar file suitable for the new switch is found on a stack member, the process extracts the file and automatically upgrades the new switch.

The auto-upgrade (auto-copy and auto-extract) processes start a few minutes after the mismatched software is detected.

When the auto-upgrade process is complete, the new switch reloads and joins the stack as a fully functioning member. If you have both stack cables connected during the reload, network downtime does not occur because the switch stack operates on two rings.

Auto-Advise

Automatic advise (auto-advise) occurs when the auto-upgrade process cannot find appropriate stack member software to copy to the new switch. This process tells you the command (**archive copy-sw** or **archive download-sw** privileged EXEC command) and the image name (tar filename) needed to manually upgrade the switch stack or the new switch. The recommended image can be the running switch stack image or a tar file in any flash file system in the switch stack (including the new switch). If an appropriate image is not found in the stack flash file systems, the auto-advise process tells you to install new software on the switch stack. Auto-advise cannot be disabled, and there is no command to check its status.

Examples of Auto-Advise Messages

When you add a switch that has a different minor version number to the switch stack, the software displays messages in sequence (assuming that there are no other system messages generated by the switch).

This example shows that the switch stack detected a new switch that is running a different minor version number than the switch stack. Auto-copy starts, finds suitable software to copy from a stack member to the switch in VM mode, upgrades the switch in VM mode, and then reloads it:

```
*Mar 11 20:31:19.247:%STACKMGR-6-STACK_LINK_CHANGE:Stack Port 2 Switch 2 has changed to
state UP
*Mar 11 20:31:23.232:%STACKMGR-6-SWITCH_ADDED_VM:Switch 1 has been ADDED to the
stack(VERSION_MISMATCH)
*Mar 11 20:31:23.291:%STACKMGR-6-SWITCH_ADDED_VM:Switch 1 has been ADDED to the
stack(VERSION_MISMATCH) (Stack 1-3)
*Mar 11 20:33:23.248:%IMAGEMGR-6-AUTO_COPY_SW_INITIATED:Auto-copy-software process initiated
for switch number(s) 1
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:Searching for stack member to act
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:as software donor...
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:Found donor (system #2) for
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:member(s) 1
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:System software to be uploaded:
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:System Type: 0x00000000
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:archiving c2960x-universalk9-mz.150-2.EX1
(directory)
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:archiving c2960x-universalk9-mz.150-2.EX1.bin
(4945851 bytes)
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:archiving
c2960x-universalk9-mz.150-2.EX1/info(450 bytes)
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:archiving info (104 bytes)
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:examining image...
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:extracting info (104 bytes)
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:extracting
c2960x-universalk9-mz.150-2.EX1/info(450 bytes)
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:extracting info (104 bytes)
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:Stacking Version Number:1.4
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:System Type: 0x00000000
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW: Ios Image File Size: 0x004BA200
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW: Total Image File Size:0x00818A00
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW: Minimum Dram required:0x08000000
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW: Image Suffix:universalk9-mz.150-2.EX1
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW: Image Directory:c2960x-universalk9-mz.150-2.EX1
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW: Image Name:c2960x-universalk9-mz.150-2.EX1
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW: Image 1:flash1:c2960x-universalk9-mz.150-2.EX1
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW: Old image will be deleted after download.
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:Extracting images from archive into flash on
switch 1...
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:c2960x-universalk9-mz.150-2.EX1 (directory)
```

```

*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:extracting
c2960x-universalk9-mz.150-2.EX1/c2960x-universalk9-mz.150-2.EX1 (4945851 bytes)
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:extracting c2960x-universalk9-mz.150-2.EX1/info
(450 bytes)
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:extracting info (104 bytes)
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:Installing
(renaming):`flash1:update/c2960x-universalk9-mz.150-2.EX1' ->
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW: `flash1:c2960x-universalk9-mz.150-2.EX1'
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:New software image installed in
flash1:c2960x-universalk9-mz.150-2.EX1
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:Removing old
image:flash1:c2960x-universalk9-mz.150-2.EX1
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:All software images installed.
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:Requested system reload in progress...
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:Software successfully copied to
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:system(s) 1
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:Done copying software
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:Reloading system(s) 1

```

This example shows that the switch stack detected a new switch that is running a different minor version number than the switch stack. Auto-copy starts but cannot find software in the switch stack to copy to the VM-mode switch to make it compatible with the switch stack. The auto-advise process starts and recommends that you download a tar file from the network to the switch in VM mode:

```

*Mar 1 00:01:11.319:%STACKMGR-6-STACK_LINK_CHANGE:Stack Port 2 Switch 2 has changed to state
UP
*Mar 1 00:01:15.547:%STACKMGR-6-SWITCH_ADDED_VM:Switch 1 has been ADDED to the stack
(VERSION_MISMATCH)
stack_2#
*Mar 1 00:03:15.554:%IMAGEMGR-6-AUTO_COPY_SW_INITIATED:Auto-copy-software process initiated
for switch number(s) 1
*Mar 1 00:03:15.554:%IMAGEMGR-6-AUTO_COPY_SW:
*Mar 1 00:03:15.554:%IMAGEMGR-6-AUTO_COPY_SW:Searching for stack member to act
*Mar 1 00:03:15.554:%IMAGEMGR-6-AUTO_COPY_SW:as software donor...
*Mar 1 00:03:15.554:%IMAGEMGR-6-AUTO_COPY_SW:Software was not copied
*Mar 1 00:03:15.562:%IMAGEMGR-6-AUTO_ADVISE_SW_INITIATED:Auto-advise-software process
initiated for switch number(s) 1
*Mar 1 00:04:22.537:%IMAGEMGR-6-AUTO_ADVISE_SW:
*Mar 1 00:04:22.537:%IMAGEMGR-6-AUTO_ADVISE_SW:
*Mar 1 00:04:22.537:%IMAGEMGR-6-AUTO_ADVISE_SW:Systems with incompatible software
*Mar 1 00:04:22.537:%IMAGEMGR-6-AUTO_ADVISE_SW:have been added to the stack. The
*Mar 1 00:04:22.537:%IMAGEMGR-6-AUTO_ADVISE_SW:storage devices on all of the stack
*Mar 1 00:04:22.537:%IMAGEMGR-6-AUTO_ADVISE_SW:members have been scanned, and it has
*Mar 1 00:04:22.537:%IMAGEMGR-6-AUTO_ADVISE_SW:been determined that the stack can be
*Mar 1 00:04:22.537:%IMAGEMGR-6-AUTO_ADVISE_SW:repaired by issuing the following
*Mar 1 00:04:22.537:%IMAGEMGR-6-AUTO_ADVISE_SW:command(s) :
*Mar 1 00:04:22.537:%IMAGEMGR-6-AUTO_ADVISE_SW:
*Mar 1 00:04:22.537:%IMAGEMGR-6-AUTO_ADVISE_SW: archive download-sw /force-reload /overwrite
/dest 1 flash1:c2960x-universalk9-mz.150-2.EX1.tar
*Mar 1 00:04:22.537:%IMAGEMGR-6-AUTO_ADVISE_SW:

```



Note Auto-advise and auto-copy identify which images are running by examining the info file and by searching the directory structure on the switch stack. If you download your image by using the **copy tftp:** boot loader command instead of the **archive download-sw** privileged EXEC command, the proper directory structure is not created.

SDM Template Mismatch in Switch Stacks

All stack members use the Switch Database Management (SDM) template configured on the active switch. When a new switch is added to a stack, the SDM configuration that is stored on the active switch overrides the template configured on an individual switch.

You can use the **show switch** privileged EXEC command to see if any stack members are in SDM-mismatch mode.

Version-mismatch (VM) mode has priority over SDM-mismatch mode. If a VM-mode condition and an SDM-mismatch mode exist, the switch stack first attempts to resolve the VM-mode condition.

For more information about SDM templates, see the *Catalyst 2960-XR Switch System Management Configuration Guide*.

Switch Stack Management Connectivity

You manage the switch stack and the stack member interfaces through the active switch. You can use the CLI, SNMP, and any of the supported network management applications. You cannot manage stack members on an individual device basis.

Connectivity to Specific Stack Members

If you want to configure a specific stack member port, you must include the stack member number in the CLI command interface notation.

To debug a specific stack member, you can access it from the active stack by using the **session stack-member-number** privileged EXEC command. The stack member number is appended to the system prompt. For example, Switch-2# is the prompt in privileged EXEC mode for stack member 2, and the system prompt for the active stack is Switch. Only the **show** and **debug** commands are available in a CLI session to a specific stack member.

Connectivity to the Switch Stack Through an IP Address

The switch stack is managed through a single IP address. The IP address is a system-level setting and is not specific to the active stack or to any other stack member. You can still manage the stack through the same IP address even if you remove the active stack or any other stack member from the stack, provided there is IP connectivity.



Note Stack members retain their IP addresses when you remove them from a switch stack. To avoid a conflict by having two devices with the same IP address in your network, change the IP addresses of any active stack that you remove from the switch stack.

For related information about switch stack configurations, see the *Switch Stack Configuration Files* section.

Connectivity to the Switch Stack Through Console Ports or Ethernet Management Ports

You can connect to the active switch by using one of these methods:

- You can connect a terminal or a PC to the active switch through the console port of one or more stack members.
- You can connect a PC to the active switch through the Ethernet management ports of one or more stack members. For more information about connecting to the switch stack through Ethernet management ports, see the *Using the Ethernet Management Port* section.

You can connect to the active switch by connecting a terminal or a PC to the active switch through the console port of one or more stack members.

When you use the console port of a stack member, a VTY session is created with the IP address in the 192.168.0.1/24 subnet.

Be careful when using multiple CLI sessions to the active switch. Commands that you enter in one session are not displayed in the other sessions. Therefore, it is possible that you might not be able to identify the session from which you entered a command.

We recommend using only one CLI session when managing the switch stack.

How to Configure a Switch Stack

Enabling the Persistent MAC Address Feature



Note When you enter the command to configure this feature, a warning message appears with the consequences of your configuration. You should use this feature cautiously. Using the old active switch MAC address elsewhere in the same domain could result in lost traffic.

Follow these steps to enable persistent MAC address:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example:	Enters global configuration mode.

	Command or Action	Purpose
	Device# <code>configure terminal</code>	
Step 3	<p>stack-mac persistent timer [0 <i>time-value</i>]</p> <p>Example:</p> <pre>Device(config)# stack-mac persistent timer 7</pre>	<p>Enables a time delay after a stack-active switch change before the stack MAC address changes to that of the new ac. If the previous active switch rejoins the stack during this period, the stack uses that MAC address as the stack MAC address.</p> <p>You can configure the time period as 0 to 60 minutes.</p> <ul style="list-style-type: none"> • Enter the command with no value to set the default delay of approximately 4 minutes. We recommend that you always enter a value. <p>If the command is entered without a value, the time delay appears in the running-config file with an explicit timer value of 4 minutes.</p> <ul style="list-style-type: none"> • Enter 0 to continue using the MAC address of the current active switch indefinitely. <p>The stack MAC address of the previous active switch is used until you enter the no stack-mac persistent timer command, which immediately changes the stack MAC address to that of the current active switch.</p> <ul style="list-style-type: none"> • Enter a <i>time-value</i> from 1 to 60 minutes to configure the time period before the stack MAC address changes to the new active switch. <p>The stack MAC address of the previous active switch is used until the configured time period expires or until you enter the no stack-mac persistent timer command.</p> <p>Note If you enter the no stack-mac persistent timer command after a new active switch takes over, before the time expires, the switch stack moves to the current active switch MAC address.</p>
Step 4	<p>end</p> <p>Example:</p>	Returns to privileged EXEC mode.

	Command or Action	Purpose
	Device(config)# end	
Step 5	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

What to do next

Use the **no stack-mac persistent timer** global configuration command to disable the persistent MAC address feature.

Assigning a Stack Member Number

This optional task is available only from the active stack.

Follow these steps to assign a member number to a stack member:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	switch <i>current-stack-member-number</i> renumber <i>new-stack-member-number</i> Example: Device(config)# switch 3 renumber 4	Specifies the current stack member number and the new stack member number for the stack member. The range is 1 to 8. You can display the current stack member number by using the show switch user EXEC command.
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 5	reload slot <i>stack-member-number</i> Example: Device# reload slot 4	Resets the stack member.
Step 6	show switch Example: showDevice	Verify the stack member number.
Step 7	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Setting the Stack Member Priority Value

This optional task is available only from the active stack.

Follow these steps to assign a priority value to a stack member:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	switch <i>stack-member-number</i> priority <i>new-priority-number</i> Example: Device# switch 3 priority 2	Specifies the stack member number and the new priority for the stack member. The stack member number range is 1 to 8. The priority value range is 1 to 15. You can display the current priority value by using the show switch user EXEC command. The new priority value takes effect immediately but does not affect the current active stack. The new priority value helps determine which stack member is elected as the new active stack when the current active stack or switch stack resets.
Step 3	show switch <i>stack-member-number</i> Example: Device# show switch	Verify the stack member priority value.

	Command or Action	Purpose
Step 4	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Provisioning a New Member for a Switch Stack

This optional task is available only from the active switch.

Procedure

	Command or Action	Purpose
Step 1	show switch Example: Device# <code>show switch</code>	Displays summary information about the switch stack.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	switch <i>stack-member-number</i> provision <i>type</i> Example: Device(config)# <code>switch 3 provision WS-xxxx</code>	Specifies the stack member number for the preconfigured switch. By default, no switches are provisioned. For <i>stack-member-number</i> , the range is 1 to 8. Specify a stack member number that is not already used in the switch stack. See Step 1. For <i>type</i> , enter the model number of a supported switch that is listed in the command-line help strings.
Step 4	end Example: Device(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 5	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Removing Provisioned Switch Information

Before you begin, you must remove the provisioned switch from the stack. This optional task is available only from the active stack.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 2	no switch <i>stack-member-number</i> provision Example: Device(config)# <code>no switch 3 provision</code>	Removes the provisioning information for the specified member.
Step 3	end Example: Device(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 4	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Example

If you are removing a provisioned switch in a stack with this configuration:

- The stack has four members
- Stack member 1 is the active stack
- Stack member 3 is a provisioned switch

and want to remove the provisioned information and to avoid receiving an error message, you can remove power from stack member 3, disconnect the stack cables between the stack member 3 and switches to which it is connected, reconnect the cables between the remaining stack members, and enter the **no switch *stack-member-number* provision** global configuration command.

Troubleshooting the Switch Stack

Accessing the CLI of a Specific Member

This optional task is for debugging purposes, and is available only from the active switch.

You can access all or specific members by using the **remote command** `{all | stack-member-number}` privileged EXEC command. The stack member number range is 1 to 8.

You can access specific members by using the **session** `stack-member-number` privileged EXEC command. The member number is appended to the system prompt. For example, the prompt for member 2 is **Switch-2#**, and system prompt for the active switch is **Switch#**. Enter **exit** to return to the CLI session on the active switch. Only the **show** and **debug** commands are available on a specific member.

Temporarily Disabling a Stack Port

If a stack port is flapping and causing instability in the stack ring, to disable the port, enter the **switch** `stack-member-number stack port port-number disable` privileged EXEC command. To reenables the port, enter the **switch** `stack-member-number stack port port-number enable` command.



Note Be careful when using the **switch** `stack-member-number stack port port-number disable` command. When you disable the stack port, the stack operates at half bandwidth.

A stack is in the full-ring state when all members are connected through the stack ports and are in the ready state.

The stack is in the partial-ring state when the following occurs:

- All members are connected through their stack ports but some are not in the ready state.
- Some members are not connected through the stack ports.

Procedure

	Command or Action	Purpose
Step 1	switch <code>stack-member-number stack port port-number disable</code> Example: Device# <code>switch 2 stack port 1 disable</code>	Disables the specified stack port.
Step 2	switch <code>stack-member-number stack port port-number enable</code> Example: Device# <code>switch 2 stack port 1 enable</code>	Reenables the stack port.

When you disable a stack port and the stack is in the full-ring state, you can disable only one stack port. This message appears:

```
Enabling/disabling a stack port may cause undesired stack changes. Continue?[confirm]
```

When you disable a stack port and the stack is in the partial-ring state, you cannot disable the port. This message appears:

```
Disabling stack port not allowed with current stack configuration.
```

Reenabling a Stack Port While Another Member Starts

Stack Port 1 on Switch 1 is connected to Port 2 on Switch 4. If Port 1 is flapping, you can disable Port 1 with the **switch 1 stack port 1 disable** privileged EXEC command. While Port 1 on Switch 1 is disabled and Switch 1 is still powered on, follow these steps to reenoble a stack port:

Procedure

-
- Step 1** Disconnect the stack cable between Port 1 on Switch 1 and Port 2 on Switch 4.
 - Step 2** Remove Switch 4 from the stack.
 - Step 3** Add a switch to replace Switch 4 and assign it switch-number 4.
 - Step 4** Reconnect the cable between Port 1 on Switch 1 and Port 2 on Switch 4 (the replacement switch).
 - Step 5** Reenable the link between the switches. Enter the **switch 1 stack port 1 enable** privileged EXEC command to enable Port 1 on Switch 1.
 - Step 6** Power on Switch 4.
-



Caution Powering on Switch 4 before enabling the Port 1 on Switch 1 might cause one of the switches to reload. If Switch 4 is powered on first, you might need to enter the **switch 1 stack port 1 enable** and the **switch 4 stack port 2 enable** privileged EXEC commands to bring up the link.

Monitoring the Device Stack

Table 2: Commands for Displaying Stack Information

Command	Description
show controller ethernet-controller stack port {1 2}	Displays stack port counters (or per-interface and per-stack port send and receive statistics read from the hardware).
show controller ethernet-controller fastethernet0	Displays information about the Ethernet management port, including the port status and the per-interface send and receive statistics read from the hardware.
show platform stack compatibility	Displays information about HULC feature compatibility.

Command	Description
show platform stack manager all	Displays all stack manager information, such as the stack protocol version.
show platform stack passive-links	Displays information about stack passive links.
show switch	Displays summary information about the stack, including the status of provisioned switches and switches in version-mismatch mode.
show switch <i>stack-member-number</i>	Displays information about a specific member.
show switch detail	Displays detailed information about the stack.
show switch neighbors	Displays the stack neighbors.
show switch stack-ports	Displays port information for the stack.

Configuration Examples for Switch Stacks

Switch Stack Configuration Scenarios

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

Table 3: Configuration Scenarios

Scenario		Result
Active switch election specifically determined by existing active switches	Connect two powered-on switch stacks through the stack ports.	Only one of the two active switches becomes the new active switch.
Active switch election specifically determined by the stack member priority value	<ol style="list-style-type: none"> 1. Connect two switches through their stack ports. 2. Use the switch <i>stack-member-number</i> priority <i>new-priority-number</i> global configuration command to set one stack member with a higher member priority value. 3. Restart both stack members at the same time. 	The stack member with the higher priority value is elected active switch.

Scenario		Result
Active switch election specifically determined by the configuration file	Assuming that both stack members have the same priority value: <ol style="list-style-type: none"> 1. Make sure that one stack member has a default configuration and that the other stack member has a saved (nondefault) configuration file. 2. Restart both stack members at the same time. 	The stack member with the saved configuration file is elected active switch.
Active switch election specifically determined by the MAC address	Assuming that both stack members have the same priority value, configuration file, and feature set, restart both stack members at the same time.	The stack member with the lower MAC address is elected active switch.
Stack member number conflict	Assuming that one stack member has a higher priority value than the other stack member: <ol style="list-style-type: none"> 1. Ensure that both stack members have the same stack member number. If necessary, use the switch <i>current-stack-member-number</i> renumber <i>new-stack-member-number</i> global configuration command. 2. Restart both stack members at the same time. 	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"> 1. Power off the new switch. 2. Through their stack ports, connect the new switch to a powered-on switch stack. 3. Power on the new switch. 	The active switch is retained. The new switch is added to the switch stack.
Active switch failure	Remove (or power off) the active switch.	One of the remaining stack members becomes the new active switch. All other stack members in the stack remain as stack members and do not reboot.

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

```
Device(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.
Device(config)# end
Device# show switch
Switch/Stack Mac Address : 0016.4727.a900
Mac persistency wait time: 7 mins

Switch#  Role  Mac Address      Priority  H/W  Current
-----  ---  -
*1      Master  0016.4727.a900    1        P2B  Ready
```

Provisioning a New Member for a Switch Stack: Example

This example shows how to provision a switch with a stack member number of 2 for the switch stack. The **show running-config** command output shows the interfaces associated with the provisioned switch:

```
Device(config)# switch 2 provision switch_PID
Device(config)# end
Device# show running-config | include switch 2
switch 2 provision switch_PID
```

Additional References for Switch Stacks

Related Documents

Related Topic	Document Title
Cabling and powering on a switch stack.	<i>Catalyst 2960-XR Switch Hardware Installation Guide</i> http://www.cisco.com/c/en/us/td/docs/switches/lan/catalyst2960cx_3650cx/hardware/installation/guide/b_2960cx-3560cx_hig.html

Error Message Decoder

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

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 software images, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

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>	http://www.cisco.com/support