



## Stack Manager Commands

---

- [debug platform remote-commands](#), on page 2
- [debug platform stack-manager](#), on page 3
- [reload](#), on page 4
- [remote command](#), on page 6
- [session](#), on page 7
- [show platform stack compatibility configuration](#), on page 8
- [show platform stack compatibility feature](#), on page 9
- [show platform stack compatibility table](#), on page 11
- [show platform stack manager](#), on page 13
- [show switch](#), on page 15
- [stack-mac persistent timer](#), on page 16
- [switch stack port](#), on page 17
- [switch priority](#), on page 18
- [switch provision](#), on page 19
- [switch renumber](#), on page 21

# debug platform remote-commands

To enable debugging of remote commands, use the **debug platform remote-commands** command in privileged EXEC mode. To disable debugging, use the **no** form of this command.

**debug platform remote-commands**  
**no debug platform remote-commands**

<b>Syntax Description</b>	This command has no arguments or keywords.
---------------------------	--

<b>Command Default</b>	Debugging is disabled.
------------------------	------------------------

<b>Command Modes</b>	Privileged EXEC
----------------------	-----------------

<b>Command History</b>	<b>Release</b> <b>Modification</b>
	This command was introduced.

<b>Usage Guidelines</b>	The <b>undebug platform remote-commands</b> command is the same as the <b>no debug platform remote-commands</b> command.
-------------------------	--

When you enable debugging on a switch stack, it is enabled only on the active switch. To enable debugging on a stack member, you can start a session from the stack's active switch by using the **session** *switch-number* privileged EXEC command. Enter the **debug** command at the command-line prompt of the stack member. You also can use the **remote command** *stack-member-number* *LINE* privileged EXEC command on the active switch to enable debugging on a member switch without first starting a session.

# debug platform stack-manager

To enable debugging of the stack manager software, use the **debug platform stack-manager** command in privileged EXEC mode. To disable debugging, use the **no** form of this command.

**debug platform stack-manager** {all | rpc | sdp | sim | ssm | trace}  
**no debug platform stack-manager** {all | rpc | sdp | sim | ssm | trace}

## Syntax Description

<b>all</b>	Displays all stack manager debug messages.
<b>rpc</b>	Displays stack manager remote procedure call (RPC) usage debug messages.
<b>sdp</b>	Displays the Stack Discovery Protocol (SDP) debug messages.
<b>sim</b>	Displays the stack information module debug messages.
<b>ssm</b>	Displays the stack state-machine debug messages.
<b>trace</b>	Traces the stack manager entry and exit debug messages.

## Command Default

Debugging is disabled.

## Command Modes

Privileged EXEC

## Command History

### Release Modification

This command was introduced.

## Usage Guidelines

This command is supported only on stacking-capable switches.

The **undebg platform stack-manager** command is the same as the **no debug platform stack-manager** command.

When you enable debugging on a switch stack, it is enabled only on the active switch. To enable debugging on a stack member, you can start a session from the active switch by using the **session switch-number EXEC** command. Enter the **debug** command at the command-line prompt of the stack member. You also can use the **remote command stack-member-number LINE EXEC** command on the active switch to enable debugging on a member switch without first starting a session.

# reload

To reload the stack member and to apply a configuration change, use the **reload** command in privileged EXEC mode.

**reload** [{/noverify | /verify}] [{*LINE* | at | cancel | in | slot *stack-member-number* | standby-cpu}]

Syntax Description	<b>/noverify</b>	(Optional) Specifies to not verify the file signature before the reload.
	<b>/verify</b>	(Optional) Verifies the file signature before the reload.
	<i>LINE</i>	(Optional) Reason for the reload.
	<b>at</b>	(Optional) Specifies the time in hh:mm for the reload to occur.
	<b>cancel</b>	(Optional) Cancels the pending reload.
	<b>in</b>	(Optional) Specifies a time interval for reloads to occur.
	<b>slot</b>	(Optional) Saves the changes on the specified stack member and then restarts it.
	<i>stack-member-number</i>	
	<b>standby-cpu</b>	(Optional) Reloads the standby route processor (RP).

**Command Default** Immediately reloads the stack member and puts a configuration change into effect.

**Command Modes** Privileged EXEC

Command History	Release	Modification
		This command was introduced.

**Usage Guidelines** If there is more than one switch in the switch stack, and you enter the **reload slot *stack-member-number*** command, you are not prompted to save the configuration.

## Examples

This example shows how to reload the switch stack:

```
# reload
System configuration has been modified. Save? [yes/no]: y
Proceed to reload the whole Stack? [confirm] y
```

This example shows how to reload a specific stack member:

```
# reload slot 6
Proceed with reload? [confirm] y
```

This example shows how to reload a single-switch switch stack (there is only one member switch):

```
# reload slot 3
System configuration has been modified. Save? [yes/no]: y
Proceed to reload the whole Stack? [confirm] y
```

# remote command

To monitor all or specified stack members, use the **remote command** privileged EXEC command.

**remote command** {*allstack-member-number*} *LINE*

Syntax Description	<b>all</b>	Applies to all stack members.
	<i>stack-member-number</i>	
	<i>LINE</i>	The command to execute.

**Command Modes** Privileged EXEC

**Command History**

Release	Modification
	This command was introduced.

**Usage Guidelines** The commands that you use in the *LINE* command-to-execute string (such as **debug**, **show**, or **clear**) apply to a specific stack member or to the switch stack.

## Examples

This example shows how to execute the **undebug** command on the switch stack:

```
Switch(config)# remote command all undebug all
Switch :1 :
-----
All possible debugging has been turned off
Switch :5 :
-----
All possible debugging has been turned off
Switch :7 :
-----
All possible debugging has been turned off
```

This example shows how to execute the **debug udd event** command on stack member 5:

```
Switch(config)# remote command 5 undebug all
Switch :5 :
-----
UDLD events debugging is on
```

# session

To access a specific stack member, use the **session** command in privileged EXEC mode on the active stack.

**session** *stack-member-number*

<b>Syntax Description</b>	<i>stack-member-number</i> Stack member number to access from the active switch.				
<b>Command Default</b>	None				
<b>Command Modes</b>	Privileged EXEC				
<b>Command History</b>	<table><tr><th>Release</th><th>Modification</th></tr><tr><td></td><td>This command was introduced.</td></tr></table>	Release	Modification		This command was introduced.
Release	Modification				
	This command was introduced.				
<b>Usage Guidelines</b>	<p>When you access the member, its member number is appended to the system prompt.</p> <p>Use the <b>session</b> command from the active switch to access a member.</p> <p>Use the <b>session</b> command with <b>processor 1</b> from the active or a standalone switch to access the internal controller. A standalone device is always member 1.</p>				
<b>Examples</b>	<p>This example shows how to access stack member 3:</p> <pre>Device# session 3 Device-3#</pre>				

# show platform stack compatibility configuration

To display switch stack compatibility information, use the **show platform stack compatibility configuration** command in privileged EXEC mode.

**show platform stack compatibility configuration** {**current** | **failure-log** | **mismatch** [**switch** *switch-number*]}

Syntax Description	<b>current</b>	Displays currently configured system level features.
	<b>failure-log</b>	Displays non-baseline feature configuration failure log.
	<b>mismatch</b>	Displays configured non-baseline features that are causing a mismatch.
	<b>switch</b> <i>switch-number</i>	(Optional) Displays configured non-baseline features that are causing a mismatch for the specified switch. The range is 1 to 8.
Command Default	None	
Command Modes	Privileged EXEC	
Command History	<b>Release</b>	<b>Modification</b>
	This command was introduced.	
Usage Guidelines	Use this command only when you are working directly with your technical support representative while troubleshooting a problem. Do not use this command unless your technical support representative asks you to do so.	

This example shows how to display switch stack compatibility information:

```
# show platform stack compatibility configuration current
```



# show platform stack compatibility feature

To display switch stack state machine and message trace feature compatibility information, use the **show platform stack compatibility feature** command in privileged EXEC mode.

**show platform stack compatibility feature** {all|independent [feature-id *feature-id*]|interdependent [feature-id *feature-id*]|port [feature-id *feature-id*]}

Syntax Description	<b>all</b>	Displays all non-baseline features.
	<b>independent</b>	Displays switch-level independent features.
	<b>feature-id</b> <i>feature-id</i>	(Optional) Displays switch-level independent, system-level interdependent or port-level independent features with the specified feature ID.
	<b>interdependent</b>	Displays system-level interdependent features.
	<b>port</b>	Displays port-level independent features.

**Command Default** None

**Command Modes** Privileged EXEC

**Command History**

Release	Modification
	This command was introduced.

**Usage Guidelines** The **show platform stack compatibility** commands display feature incompatibility information between stack members in a mixed stack. Compatibility checks ensure that features that are supported across members of a stack using a forwarding ASIC with different capabilities will function effortlessly and error free.

Use this command only when you are working directly with your technical support representative while troubleshooting a problem. Do not use this command unless your technical support representative asks you to do so.

This example shows how to display all non-baseline features:

```
# show platform stack compatibility feature all
      System Level Interdependent Features
=====
1: FHRR hardware vlan entry sharing feature
2: Jumbo MTU Routing Support
3: VRF on PVLAN interface
4: Global VRF config with greater than MAX policies
5: Routing keyword in IPv6 ACL
6: Ahp keyword in IPv6 ACL
7: Unsupported prefixes in IPv6 ACL
8: Unicast Reverse Path Forwarding
9: Multiple FHRP support
10: Gateway Load Balancing ProtocolVLAN-based FSPAN
11: CTS tagging and role-based enforcement
12: SPAN support 4 source session
13: Unknown
```

## Switch Level Independent Features

=====

## Port Level Independent Features

=====

- 1: Routing keyword in IPv6 ACL
- 2: Ahp keyword in IPv6 ACL
- 3: Unsupported prefixes in IPv6 ACL
- 4: Port-based FSPAN
- 5: IPv6 QoS match protocol support
- 6: IPv6 QoS ipv6 named ACL support

# show platform stack compatibility table

To display feature compatibility tables for the switch stack, use the **show platform stack compatibility table** command in privileged EXEC mode.

**show platform stack compatibility table** {all | independent | interdependent | port}

<b>Syntax Description</b>	<b>all</b>	Displays all feature compatibility tables.
	<b>independent</b>	Displays a switch-level independent feature compatibility table.
	<b>interdependent</b>	Displays a system-level interdependent feature compatibility table.
	<b>port</b>	Displays a port-level independent feature compatibility table.

**Command Default** None

**Command Modes** Privileged EXEC

**Command History**

Release	Modification
	This command was introduced.

**Usage Guidelines** The **show platform stack compatibility** commands display feature incompatibility information between stack members in a mixed stack. Compatibility checks ensure that features that are supported across members of a stack using a forwarding ASIC with different capabilities will function effortlessly and error free.

Use this command only when you are working directly with your technical support representative while troubleshooting a problem. Do not use this command unless your technical support representative asks you to do so.

This example shows how to display all feature compatibility tables:

```
# show platform stack compatibility feature all
System Level Interdependent Feature Matrix
=====
1: FHRR hardware vlan entry sharing feature
   Supported by switch versions: 1 3 4
2: Jumbo MTU Routing Support
   Supported by switch versions: 1 3 4
3: VRF on PVLAN interface
   Supported by switch versions: 1 3 4
4: Global VRF config with greater than MAX policies
   Supported by switch versions: 1 3 4
5: Routing keyword in IPv6 ACL
   Supported by switch versions: 1 3 4
6: Ahp keyword in IPv6 ACL
   Supported by switch versions: 1 3 4
7: Unsupported prefixes in IPv6 ACL
   Supported by switch versions: 1 3 4
8: Unicast Reverse Path Forwarding
   Supported by switch versions: 1 3 4
9: Multiple FHRP support
   Supported by switch versions: 1 3 4
```

## show platform stack compatibility table

```

10: Gateway Load Balancing ProtocolVLAN-based FSPAN
    Supported by switch versions: 1 2 3 4
11: CTS tagging and role-based enforcement
    Supported by switch versions: 1 3 4
12: SPAN support 4 source session
    Supported by switch versions: 3
13: Unknown
    Supported by switch versions: 4

```

## Switch Level Independent Feature Matrix

```
=====
```

## Port Level Independent Feature Matrix

```
=====
```

```

1: Routing keyword in IPv6 ACL
    Supported by ASIC versions: 1 2 3
2: Ahp keyword in IPv6 ACL
    Supported by ASIC versions: 1 2 3
3: Unsupported prefixes in IPv6 ACL
    Supported by ASIC versions: 1 2 3
4: Port-based FSPAN
    Supported by ASIC versions: 1 2 3
5: IPv6 QoS match protocol support
    Supported by ASIC versions: 1 2 3
6: IPv6 QoS ipv6 named ACL support
    Supported by ASIC versions: 1 2 3

```

# show platform stack manager

To display platform-dependent switch-stack information, use the **show platform stack manager** command in privileged EXEC mode.

```
show platform stack manager {all | counters | trace [{cs [cs] | sdp [reverse] | state [reverse] | tdm}]}
```

<b>Syntax Description</b>	<b>all</b>	Displays all information for the entire switch stack.
	<b>counters</b>	Displays the stack manager counters.
	<b>trace</b>	Displays trace information.
	<b>cs</b>	(Optional) Displays information about changes in stack-related trace messages.
	<b>sdp</b>	(Optional) Displays Stack Discovery Protocol (SDP) information.
	<b>reverse</b>	(Optional) Displays trace information in reverse chronological order (from recent to older chronological sequence).
	<b>state</b>	(Optional) Displays stack state machine information.
	<b>tdm</b>	(Optional) Displays information about topology discovery including a summary of the stacking over Ethernet state machine events and messages.
<b>Command Default</b>	None	
<b>Command Modes</b>	Privileged EXEC	
<b>Command History</b>	<b>Release</b>	<b>Modification</b>
	This command was introduced.	

**Usage Guidelines** Use this command only when you are working directly with your technical support representative while troubleshooting a problem. Do not use this command unless your technical support representative asks you to do so.



**Note** This command is supported only on stacking-capable switches.

The summary information about the switch stack shows these states:

- **Waiting**—A switch is booting up and waiting for communication from other switches in the stack. The switch has not determined whether or not it is active switch.

Stack members not participating in election remain in the waiting state until the active switch is elected and ready.

- **Initializing**—A switch has determined whether it is the active switch. If not, the switch receives its system- and interface-level configuration from the active switch and loads it.
- **Ready**—The member has completed loading the system- and interface-level configurations and can forward traffic.
- **Ver Mismatch**—A switch in version mismatch mode. Version-mismatch mode is when a switch that joins the stack has a different stack protocol minor version number than the active switch.

A typical state transition for a stack member (including the active) booting up is Waiting > Initializing > Ready.

A typical state transition for a stack member to active switch after an election is Ready > Re-Init > Ready.

A typical state transition for a stack member in version mismatch (VM) mode is Waiting > Ver Mismatch.

# show switch

To display information that is related to the stack member or the switch stack, use the **show switch** command in EXEC mode.

## Command Default

None

## Command Modes

User EXEC (>)

Privileged EXEC (#)

## Command History

### Release Modification

This command was introduced.

## Examples

This example shows how to display summary stack information:

This example shows how to display detailed stack information:

This example shows how to display the member 6 summary information:

```
Device# show switch 6
Switch#  Role      Mac Address      Priority    State
-----
  6      Member      0003.e31a.1e00      1          Ready
```

This example shows how to display the neighbor information for a stack:

```
Device# show switch neighbors
Switch #   Port A   Port B
-----
  6         None      8
  8         6       None
```

This example shows how to display stack-port information:

```
Device# show switch stack-ports
Switch #   Port A   Port B
-----
  6        Down      Ok
  8         Ok      Down
```

## stack-mac persistent timer

To enable the persistent MAC address feature, use the **stack-mac persistent timer** command in global configuration mode on the switch stack or on a standalone switch. To disable the persistent MAC address feature, use the **no** form of this command.

**stack-mac persistent timer** [{0*time-value*}]  
**no stack-mac persistent timer**

<b>Syntax Description</b>	<p><b>0</b></p> <hr/> <p><i>time-value</i> (Optional) Time period in minutes before the stack MAC address changes to that of the new active switch. The range is 1 to 60 minutes.</p> <hr/>				
<b>Command Default</b>	Persistent MAC address is disabled. The MAC address of the stack is always that of the first active switch.				
<b>Command Modes</b>	Global configuration (config)				
<b>Command History</b>	<table> <tr> <th>Release</th><th>Modification</th></tr> <tr> <td></td><td>This command was introduced.</td></tr> </table>	Release	Modification		This command was introduced.
Release	Modification				
	This command was introduced.				



# switch stack port

To disable or enable the specified stack port on the member, use the **switch** command in privileged EXEC mode on a stack member.

**switch** *stack-member-number* **stack port** *port-number* {**disable** | **enable**}

## Syntax Description

*stack-member-number*

**stack port** *port-number* Specifies the stack port on the member. The range is 1 to 2.

**disable** Disables the specified port.

**enable** Enables the specified port.

## Command Default

The stack port is enabled.

## Command Modes

Privileged EXEC

## Command History

**Release** **Modification**

This command was introduced.

## Usage Guidelines

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.



### 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.

If you enter the **switch** *stack-member-number* **stack port** *port-number* **disable** privileged EXEC command 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]
```

If you enter the **switch** *stack-member-number* **stack port** *port-number* **disable** privileged EXEC command 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.
```

## Examples

This example shows how to disable stack port 2 on member 4:

```
# switch 4 stack port 2 disable
```

# switch priority

To change the stack member priority value, use the **switch priority** command in mode on the active switch.

**switch** *stack-member-number* **priority** *new-priority-value*

## Syntax Description

*stack-member-number*

*new-priority-value*      New stack member priority value. The range is 1 to 15.

## Command Default

The default priority value is 1.

## Command History

### Release    Modification

This command was introduced.

## Usage Guidelines

The new priority value is a factor when a new active switch is elected. When you change the priority value, the active switch is not changed immediately.

## Examples

This example shows how to change the priority value of stack member 6 to 8:

```
Device switch 6 priority 8
Changing the Switch Priority of Switch Number 6 to 8
Do you want to continue?[confirm]
```

# switch provision

To supply a configuration to a new switch before it joins the switch stack, use the **switch provision** command in global configuration mode on the active switch. To delete all configuration information that is associated with the removed switch (a stack member that has left the stack), use the **no** form of this command.

**switch** *stack-member-number* **provision** *type*  
**no switch** *stack-member-number* **provision**

<b>Syntax Description</b>	<div><i>stack-member-number</i></div> <div><i>type</i> Switch type of the new switch before it joins the stack.</div>				
<b>Command Default</b>	The switch is not provisioned.				
<b>Command Modes</b>	Global configuration (config)				
<b>Command History</b>	<table> <tr> <th data-bbox="378 842 618 873">Release</th><th data-bbox="618 842 1539 873">Modification</th></tr> <tr> <td></td><td data-bbox="472 894 805 926">This command was introduced.</td></tr> </table>	Release	Modification		This command was introduced.
Release	Modification				
	This command was introduced.				
<b>Usage Guidelines</b>	<p>For <i>type</i>, enter the model number of a supported switch that is listed in the command-line help strings.</p> <p>To avoid receiving an error message, you must remove the specified switch from the switch stack before using the <b>no</b> form of this command to delete a provisioned configuration.</p> <p>To change the switch type, you must also remove the specified switch from the switch stack. You can change the stack member number of a provisioned switch that is physically present in the switch stack if you do not also change the switch type.</p> <p>If the switch type of the provisioned switch does not match the switch type in the provisioned configuration on the stack, the switch stack applies the default configuration to the provisioned switch and adds it to the stack. The switch stack displays a message when it applies the default configuration.</p> <p>Provisioned information appears in the running configuration of the switch stack. When you enter the <b>copy running-config startup-config</b> privileged EXEC command, the provisioned configuration is saved in the startup configuration file of the switch stack.</p>				



## Caution

When you use the **switch provision** command, memory is allocated for the provisioned configuration. When a new switch type is configured, the previously allocated memory is not fully released. Therefore, do not use this command more than approximately 200 times, or the switch will run out of memory and unexpected behavior will result.

## Examples

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.

```
(config)# switch 2 provision WS-xxxx
(config)# end
```

```
# show running-config | include switch 2
!
interface GigabitEthernet2/0/1
!
interface GigabitEthernet2/0/2
!
interface GigabitEthernet2/0/3
<output truncated>
```

You also can enter the **show switch** user EXEC command to display the provisioning status of the switch stack.

This example shows how to delete all configuration information about stack member 5 when the switch is removed from the stack:

```
(config)# no switch 5 provision
```

You can verify that the provisioned switch is added to or removed from the running configuration by entering the **show running-config** privileged EXEC command.

# switch renumber

To change the stack member number, use the **switch renumber** command in mode on the active switch.

**switch** *current-stack-member-number* **renumber** *new-stack-member-number*

## Syntax Description

*current-stack-member-number*

*new-stack-member-number*

## Command Default

The default stack member number is 1.

## Command History

### Release Modification

This command was introduced.

## Usage Guidelines

If another stack member is already using the member number that you just specified, the active switch assigns the lowest available number when you reload the stack member.



### Note

If you change the number of a stack member, and no configuration is associated with the new stack member number, that stack member loses its current configuration and resets to its default configuration.

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

Use the **reload slot** *current stack member number* privileged EXEC command to reload the stack member and to apply this configuration change.

## Examples

This example shows how to change the member number of stack member 6 to 7:

 switch renumber