Interface and Hardware Commands

- debug ilpower, on page 3
- debug interface, on page 4
- debug lldp packets, on page 5
- debug platform poe, on page 6
- debug platform software fed switch active punt packet-capture start, on page 7
- duplex, on page 8
- errdisable detect cause, on page 10
- errdisable recovery cause, on page 12
- errdisable recovery interval, on page 14
- interface, on page 15
- interface range, on page 17
- ip mtu, on page 19
- ipv6 mtu, on page 20
- lldp (interface configuration), on page 21
- logging event power-inline-status, on page 23
- mdix auto, on page 24
- mode (power-stack configuration), on page 25
- network-policy, on page 27
- network-policy profile (global configuration), on page 28
- power efficient-ethernet auto, on page 29
- power-priority, on page 30
- power inline, on page 32
- power inline police, on page 35
- power supply, on page 37
- show eee, on page 39
- show env, on page 42
- show errdisable detect, on page 45
- show errdisable recovery, on page 46
- show interfaces, on page 47
- show interfaces counters, on page 51
- show interfaces switchport, on page 53
- show interfaces transceiver, on page 55
- show memory platform, on page 59
• show module, on page 62
• show mgmt-infra trace messages ilpower, on page 63
• show mgmt-infra trace messages ilpower-ha, on page 65
• show mgmt-infra trace messages platform-mgr-poe, on page 66
• show network-policy profile, on page 67
• show platform hardware fed switch forward, on page 68
• show platform hardware fed switch forward interface, on page 71
• show platform hardware fed switch forward last summary, on page 74
• show platform resources, on page 77
• show platform software fed switch punt cpu rates, on page 78
• show platform software fed switch punt packet-capture display, on page 80
• show platform software fed switch punt rates interfaces, on page 82
• show platform software ilpower, on page 85
• show platform software memory, on page 87
• show platform software process list, on page 93
• show platform software process memory, on page 97
• show platform software process slot switch, on page 100
• show platform software status control-processor, on page 102
• show platform software thread list, on page 105
• show processes cpu platform, on page 107
• show processes cpu platform history, on page 110
• show processes cpu platform monitor, on page 113
• show processes memory platform, on page 115
• show processes platform, on page 119
• show power inline, on page 122
• show stack-power, on page 128
• show system mtu, on page 130
• show tech-support, on page 131
• show tech-support diagnostic, on page 133
• speed, on page 138
• stack-power, on page 140
• switchport block, on page 142
• system mtu, on page 143
• test mcu read-register, on page 144
• voice-signaling vlan (network-policy configuration), on page 146
• voice vlan (network-policy configuration), on page 148
debug ilpower

To enable debugging of the power controller and Power over Ethernet (PoE) system, use the `debug ilpower` command in privileged EXEC mode. To disable debugging, use the `no` form of this command.

```
debug ilpower {cdp | event | ha | ipc | police | port | powerman | registries | scp | sense}

no debug ilpower {cdp | event | ha | ipc | police | port | powerman | registries | scp | sense}
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>cdp</code></td>
<td>Displays PoE Cisco Discovery Protocol (CDP) debug messages.</td>
</tr>
<tr>
<td><code>event</code></td>
<td>Displays PoE event debug messages.</td>
</tr>
<tr>
<td><code>ha</code></td>
<td>Displays PoE high-availability messages.</td>
</tr>
<tr>
<td><code>ipc</code></td>
<td>Displays PoE Inter-Process Communication (IPC) debug messages.</td>
</tr>
<tr>
<td><code>police</code></td>
<td>Displays PoE police debug messages.</td>
</tr>
<tr>
<td><code>port</code></td>
<td>Displays PoE port manager debug messages.</td>
</tr>
<tr>
<td><code>powerman</code></td>
<td>Displays PoE power management debug messages.</td>
</tr>
<tr>
<td><code>registries</code></td>
<td>Displays PoE registries debug messages.</td>
</tr>
<tr>
<td><code>scp</code></td>
<td>Displays PoE SCP debug messages.</td>
</tr>
<tr>
<td><code>sense</code></td>
<td>Displays PoE sense debug messages.</td>
</tr>
</tbody>
</table>

**Command Default**

Debugging is disabled.

**Command Modes**

Privileged EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE 3.2SE</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>Cisco IOS XE 3.3SE</td>
<td>The upoe keyword was added.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

This command is supported only on PoE-capable switches.

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. Then enter the `debug` command at the command-line prompt of the stack member.
debug interface

To enable debugging of interface-related activities, use the **debug interface** command in privileged EXEC mode. To disable debugging, use the **no** form of this command.

```
depaginterface {interface-id | counters {exceptions | protocol memory} | states}
no debug interface {interface-id | counters {exceptions | protocol memory} | states}
```

<table>
<thead>
<tr>
<th>Syntax Description</th>
<th>ID of the physical interface. Displays debug messages for the specified physical port, identified by type switch number/module number/port, for example, gigabitethernet 1/0/2.</th>
</tr>
</thead>
<tbody>
<tr>
<td>interface-id</td>
<td>Displays counters debugging information.</td>
</tr>
<tr>
<td>counters</td>
<td>Displays debug messages when a recoverable exceptional condition occurs during the computation of the interface packet and data rate statistics.</td>
</tr>
<tr>
<td>exceptions</td>
<td>Displays debug messages for memory operations of protocol counters.</td>
</tr>
<tr>
<td>protocol memory</td>
<td>Displays intermediary debug messages when an interface's state transitions.</td>
</tr>
<tr>
<td>states</td>
<td></td>
</tr>
</tbody>
</table>

**Command Default**

Debugging is disabled.

**Command Modes**

Privileged EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE 3.2SE</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

If you do not specify a keyword, all debug messages appear.

The **undebug interface** command is the same as the **no debug interface** 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. Then enter the **debug** command at the command-line prompt of the stack member.
debug lldp packets

To enable debugging of Link Layer Discovery Protocol (LLDP) packets, use the `debug lldp packets` command in privileged EXEC mode. To disable debugging, use the `no` form of this command.

```
debug lldp packets
no debug lldp packets
```

**Syntax Description**
This command has no arguments or keywords.

**Command Default**
Debugging is disabled.

**Command Modes**
Privileged EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE 3.2SE</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**
The `undebug lldp packets` command is the same as the `no debug lldp packets` command.

When you enable debugging on a switch stack, it is enabled only on the. To enable debugging on a stack member, you can start a session from the by using the `session switch-number` EXEC command.
debug platform poe

To enable debugging of a Power over Ethernet (PoE) port, use the `debug platform poe` command in privileged EXEC mode. To disable debugging, use the `no` form of this command.

```
depth platform poe  [{error} | {error | info}]  [switch  switch-number]
no debug platform poe  [{error} | {error | info}]  [switch  switch-number]
```

**Syntax Description**

- **error**  
  (Optional) Displays PoE-related error debug messages.

- **info**  
  (Optional) Displays PoE-related information debug messages.

- **switch  switch-number**  
  (Optional) Specifies the stack member. This keyword is supported only on stacking-capable switches.

**Command Default**

Debugging is disabled.

**Command Modes**

Privileged EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE 3.2SE</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

The `undebug platform poe` command is the same as the `no debug platform poe` command.
**debug platform software fed switch active punt packet-capture start**

To enable debugging of packets during high CPU utilization, for an active switch, use the `debug platform software fed switch active punt packet-capture start` command in privileged EXEC mode. To disable debugging of packets during high CPU utilization, for an active switch, use the `debug platform software fed switch active punt packet-capture stop` command in privileged EXEC mode.

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>switch active</td>
<td>Displays information about the active switch.</td>
</tr>
<tr>
<td>punt</td>
<td>Specifies the punt information.</td>
</tr>
<tr>
<td>packet-capture</td>
<td>Specifies information about the captured packet.</td>
</tr>
<tr>
<td>start</td>
<td>Enables debugging of the active switch.</td>
</tr>
<tr>
<td>stop</td>
<td>Disables debugging of the active switch.</td>
</tr>
</tbody>
</table>

**Command Modes**

Privileged EXEC (#)

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE Gibraltar 16.11.1</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

The `debug platform software fed switch active punt packet-capture start` command starts the debugging of packets during high CPU utilization. The packet capture is stopped when the 4k buffer size is exceeded.

**Examples**

The following is a sample output from the `debug platform software fed switch active punt packet-capture start` command:

```
Device# debug platform software fed switch active punt packet-capture start
Punt packet capturing started.
```

The following is a sample output from the `debug platform software fed switch active punt packet-capture stop` command:

```
Device# debug platform software fed switch active punt packet-capture stop
Punt packet capturing stopped. Captured 101 packet(s)
```
To specify the duplex mode of operation for a port, use the `duplex` command in interface configuration mode. To return to the default value, use the `no` form of this command.

```
duplex  {auto | full | half}
no duplex  {auto | full | half}
```

**Syntax Description**
- `auto` Enables automatic duplex configuration. The port automatically detects whether it should run in full- or half-duplex mode, depending on the attached device mode.
- `full` Enables full-duplex mode.
- `half` Enables half-duplex mode (only for interfaces operating at 10 or 100 Mb/s). You cannot configure half-duplex mode for interfaces operating at 1000 or 10,000 Mb/s.

**Command Default**
The default is `auto` for Gigabit Ethernet ports. You cannot configure the duplex mode on 10-Gigabit Ethernet ports; it is always `full`. Duplex options are not supported on the 1000BASE-\(x\) or 10GBASE-\(x\) (where \(x\) is -BX, -CWDM, -LX, -SX, or -ZX) small form-factor pluggable (SFP) modules.

**Command Modes**
Interface configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE 3.2SE</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

For Gigabit Ethernet ports, setting the port to `auto` has the same effect as specifying `full` if the attached device does not autonegotiate the duplex parameter.

**Note**
Half-duplex mode is supported on Gigabit Ethernet interfaces if the duplex mode is `auto` and the connected device is operating at half duplex. However, you cannot configure these interfaces to operate in half-duplex mode.

Certain ports can be configured to be either full duplex or half duplex. How this command is applied depends on the device to which the switch is attached.

If both ends of the line support autonegotiation, we highly recommend using the default autonegotiation settings. If one interface supports autonegotiation and the other end does not, configure duplex and speed on both interfaces, and use the `auto` setting on the supported side.

If the speed is set to `auto`, the switch negotiates with the device at the other end of the link for the speed setting and then forces the speed setting to the negotiated value. The duplex setting remains as configured on each end of the link, which could result in a duplex setting mismatch.

You can configure the duplex setting when the speed is set to `auto`. 
Caution

Changing the interface speed and duplex mode configuration might shut down and reenable the interface during the reconfiguration.

You can verify your setting by entering the `show interfaces` privileged EXEC command.

Examples

This example shows how to configure an interface for full-duplex operation:

```
Device(config)# interface gigabitethernet1/0/1
Device(config-if)# duplex full
```
To enable error-disable detection for a specific cause or for all causes, use the `errdisable detect cause` command in global configuration mode. To disable the error-disable detection feature, use the `no` form of this command.

```
errdisable detect cause {all | arp-inspection | bpduguard shutdown vlan | dhcp-rate-limit | dtp-flap | gbic-invalid | inline-power | l2ptguard | link-flap | loopback | pagp-flap | pppoe-ia-rate-limit | security-violation shutdown vlan | sfp-config-mismatch}
no errdisable detect cause {all | arp-inspection | bpduguard shutdown vlan | dhcp-rate-limit | dtp-flap | gbic-invalid | inline-power | l2ptguard | link-flap | loopback | pagp-flap | pppoe-ia-rate-limit | security-violation shutdown vlan | sfp-config-mismatch}
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>all</code></td>
<td>Enables error detection for all error-disabled causes.</td>
</tr>
<tr>
<td><code>arp-inspection</code></td>
<td>Enables error detection for dynamic Address Resolution Protocol (ARP) inspection.</td>
</tr>
<tr>
<td><code>bpduguard shutdown vlan</code></td>
<td>Enables per-VLAN error-disable for BPDU guard.</td>
</tr>
<tr>
<td><code>dhcp-rate-limit</code></td>
<td>Enables error detection for DHCP snooping.</td>
</tr>
<tr>
<td><code>dtp-flap</code></td>
<td>Enables error detection for the Dynamic Trunking Protocol (DTP) flapping.</td>
</tr>
<tr>
<td><code>gbic-invalid</code></td>
<td>Enables error detection for an invalid Gigabit Interface Converter (GBIC) module. Note This error refers to an invalid small form-factor pluggable (SFP) module.</td>
</tr>
<tr>
<td><code>inline-power</code></td>
<td>Enables error detection for the Power over Ethernet (PoE) error-disabled cause. Note This keyword is supported only on switches with PoE ports.</td>
</tr>
<tr>
<td><code>l2ptguard</code></td>
<td>Enables error detection for a Layer 2 protocol-tunnel error-disabled cause.</td>
</tr>
<tr>
<td><code>link-flap</code></td>
<td>Enables error detection for link-state flapping.</td>
</tr>
<tr>
<td><code>loopback</code></td>
<td>Enables error detection for detected loopbacks.</td>
</tr>
<tr>
<td><code>pagp-flap</code></td>
<td>Enables error detection for the Port Aggregation Protocol (PAGP) flap error-disabled cause.</td>
</tr>
<tr>
<td><code>pppoe-ia-rate-limit</code></td>
<td>Enables error detection for the PPPoE Intermediate Agent rate-limit error-disabled cause.</td>
</tr>
<tr>
<td><code>security-violation shutdown vlan</code></td>
<td>Enables voice aware 802.1x security.</td>
</tr>
<tr>
<td><code>sfp-config-mismatch</code></td>
<td>Enables error detection on an SFP configuration mismatch.</td>
</tr>
</tbody>
</table>
Command Default

Detection is enabled for all causes. All causes, except per-VLAN error disabling, are configured to shut down the entire port.

Command Modes

Global configuration

Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE 3.2SE</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

Usage Guidelines

A cause (such as a link-flap or dhcp-rate-limit) is the reason for the error-disabled state. When a cause is detected on an interface, the interface is placed in an error-disabled state, an operational state that is similar to a link-down state.

When a port is error-disabled, it is effectively shut down, and no traffic is sent or received on the port. For the bridge protocol data unit (BPDU) guard, voice-aware 802.1x security, and port-security features, you can configure the switch to shut down only the offending VLAN on the port when a violation occurs, instead of shutting down the entire port.

If you set a recovery mechanism for the cause by entering the `errdisable recovery` global configuration command, the interface is brought out of the error-disabled state and allowed to retry the operation when all causes have timed out. If you do not set a recovery mechanism, you must enter the `shutdown` and then the `no shutdown` commands to manually recover an interface from the error-disabled state.

To verify your settings, enter the `show errdisable detect` privileged EXEC command.

This example shows how to enable error-disabled detection for the link-flap error-disabled cause:

```
Device(config)# errdisable detect cause link-flap
```

This command shows how to globally configure BPDU guard for a per-VLAN error-disabled state:

```
Device(config)# errdisable detect cause bpduguard shutdown vlan
```

This command shows how to globally configure voice-aware 802.1x security for a per-VLAN error-disabled state:

```
Device(config)# errdisable detect cause security-violation shutdown vlan
```

You can verify your setting by entering the `show errdisable detect` privileged EXEC command.
errdisable recovery cause

To enable the error-disabled mechanism to recover from a specific cause, use the **errdisable recovery cause** command in global configuration mode. To return to the default setting, use the **no** form of this command.

```
errdisable recovery cause {all | arp-inspection | bpduguard | channel-misconfig | dhcp-rate-limit |
dtp-flap | gbic-invalid | inline-power | l2ptguard | link-flap | loopback | mac-limit | pagp-flap |
port-mode-failure | pppoe-ia-rate-limit | psecure-violation | security-violation | sfp-config-mismatch |
storm-control | uddl}
no errdisable recovery cause {all | arp-inspection | bpduguard | channel-misconfig | dhcp-rate-limit |
dtp-flap | gbic-invalid | inline-power | l2ptguard | link-flap | loopback | mac-limit | pagp-flap |
port-mode-failure | pppoe-ia-rate-limit | psecure-violation | security-violation | sfp-config-mismatch |
storm-control | uddl}
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>all</td>
<td>Enables the timer to recover from all error-disabled causes.</td>
</tr>
<tr>
<td>arp-inspection</td>
<td>Enables the timer to recover from the Address Resolution Protocol (ARP) inspection error-disabled state.</td>
</tr>
<tr>
<td>bpduguard</td>
<td>Enables the timer to recover from the bridge protocol data unit (BPDU) guard error-disabled state.</td>
</tr>
<tr>
<td>channel-misconfig</td>
<td>Enables the timer to recover from the EtherChannel misconfiguration error-disabled state.</td>
</tr>
<tr>
<td>dhcp-rate-limit</td>
<td>Enables the timer to recover from the DHCP snooping error-disabled state.</td>
</tr>
<tr>
<td>dtp-flap</td>
<td>Enables the timer to recover from the Dynamic Trunking Protocol (DTP) flap error-disabled state.</td>
</tr>
<tr>
<td>gbic-invalid</td>
<td>Enables the timer to recover from an invalid Gigabit Interface Converter (GBIC) module error-disabled state. <strong>Note</strong> This error refers to an invalid small form-factor pluggable (SFP) error-disabled state.</td>
</tr>
<tr>
<td>inline-power</td>
<td>Enables the timer to recover from the Power over Ethernet (PoE) error-disabled state. This keyword is supported only on switches with PoE ports.</td>
</tr>
<tr>
<td>l2ptguard</td>
<td>Enables the timer to recover from a Layer 2 protocol tunnel error-disabled state.</td>
</tr>
<tr>
<td>link-flap</td>
<td>Enables the timer to recover from the link-flap error-disabled state.</td>
</tr>
<tr>
<td>loopback</td>
<td>Enables the timer to recover from a loopback error-disabled state.</td>
</tr>
<tr>
<td>mac-limit</td>
<td>Enables the timer to recover from the mac limit error-disabled state.</td>
</tr>
</tbody>
</table>
EnablesthetimertorecoverfromthePortAggregationProtocol (PAgP)-flap error-disabled state.

Enablesthetimertorecoverfromtheportmodechangefailure error-disabled state.

EnablesthetimertorecoverfromthePPPoE IA rate limit error-disabled state.

Enablesthetimertorecoverfromapotsecurityviolationdisable state.

EnablesthetimertorecoverfromanIEEE802.1x-violationdisabled state.

EnableserrordetectiononanSFPconfigurationmismatch.

Enablesthetimertorecoverfromastormcontrolerror.

EnablesthetimertorecoverfromtheUniDirectionalLinkDetection (UDLD) error-disabled state.

**Command Default**

Recovery is disabled for all causes.

**Command Modes**

Global configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE 3.2SE</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

A cause (such as all or BDPU guard) is defined as the reason that the error-disabled state occurred. When a cause is detected on an interface, the interface is placed in the error-disabled state, an operational state similar to link-down state.

When a port is error-disabled, it is effectively shut down, and no traffic is sent or received on the port. For the BPDU guard and port-security features, you can configure the switch to shut down only the offending VLAN on the port when a violation occurs, instead of shutting down the entire port.

If you do not enable the recovery for the cause, the interface stays in the error-disabled state until you enter the `shutdown` and the `no shutdown` interface configuration commands. If you enable the recovery for a cause, the interface is brought out of the error-disabled state and allowed to retry the operation again when all the causes have timed out.

Otherwise, you must enter the `shutdown` and then the `no shutdown` commands to manually recover an interface from the error-disabled state.

You can verify your settings by entering the `show errdisable recovery` privileged EXEC command.

**Examples**

This example shows how to enable the recovery timer for the BPDU guard error-disabled cause:

```
Device(config)# errdisable recovery cause bpdu-guard
```
errdisable recovery interval

To specify the time to recover from an error-disabled state, use the `errdisable recovery interval` command in global configuration mode. To return to the default setting, use the `no` form of this command.

```
errdisable recovery interval timer-interval
no errdisable recovery interval timer-interval
```

**Syntax Description**

- `timer-interval`: Time to recover from the error-disabled state. The range is 30 to 86400 seconds. The same interval is applied to all causes. The default interval is 300 seconds.

**Command Default**

The default recovery interval is 300 seconds.

**Command Modes**

Global configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE 3.2SE</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

The error-disabled recovery timer is initialized at a random differential from the configured interval value. The difference between the actual timeout value and the configured value can be up to 15 percent of the configured interval.

You can verify your settings by entering the `show errdisable recovery` privileged EXEC command.

**Examples**

This example shows how to set the timer to 500 seconds:

```
Device(config)# errdisable recovery interval 500
```
To configure an interface, use the **interface** command.

```
interface [Auto-Template interface-number | GigabitEthernet switch-number/slot-number/port-number |
          Group VI Group VI interface number | Internal Interface Internal Interface number | Loopback
          interface-number Null interface-number Port-channel interface-number TenGigabitEthernet
          switch-number/slot-number/port-number Tunnel interface-number Vlan interface-number ]
```

### Table: Interface Configuration Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Auto-Template</strong></td>
<td>Enables you to configure an auto-template interface. The range is from 1 to 999.</td>
</tr>
<tr>
<td><strong>GigabitEthernet</strong></td>
<td>Enables you to configure a Gigabit Ethernet IEEE 802.3z interface. The range is from 0 to 9.</td>
</tr>
<tr>
<td><strong>Group VI</strong></td>
<td>Enables you to configure a Group VI interface. The range is from 0 to 9.</td>
</tr>
<tr>
<td><strong>Internal Interface</strong></td>
<td>Enables you to configure an internal interface.</td>
</tr>
<tr>
<td><strong>Loopback</strong></td>
<td>Enables you to configure a loopback interface. The range is from 0 to 2147483647.</td>
</tr>
<tr>
<td><strong>Null</strong></td>
<td>Enables you to configure a null interface. The default value is 0.</td>
</tr>
<tr>
<td><strong>Port-channel</strong></td>
<td>Enables you to configure a port-channel interface. The range is from 1 to 128.</td>
</tr>
<tr>
<td><strong>TenGigabitEthernet</strong></td>
<td>Enables you to configure a 10-Gigabit Ethernet interface.</td>
</tr>
<tr>
<td></td>
<td>• <strong>switch-number</strong> — Switch ID. The range is from 1 to 8.</td>
</tr>
<tr>
<td></td>
<td>• <strong>slot-number</strong> — Slot number. The range is from 0 to 1.</td>
</tr>
<tr>
<td></td>
<td>• <strong>port-number</strong> — Port number. The range is from 1 to 24 and 37 to 48</td>
</tr>
<tr>
<td><strong>Tunnel</strong></td>
<td>Enables you to configure a tunnel interface. The range is from 0 to 2147483647.</td>
</tr>
<tr>
<td><strong>Vlan</strong></td>
<td>Enables you to configure a switch VLAN. The range is from 1 to 4094.</td>
</tr>
</tbody>
</table>

**Command Default**       | None                                                                 |
**Command Modes**         | Global configuration                                                  |
Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE 3.2SE</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

Usage Guidelines

You cannot use the "no" form of this command.

The following example shows how to configure a tunnel interface:

Device# interface Tunnel 15
# interface range

To configure an interface range, use the `interface range` command.

```bash
interface range {Auto-Template interface-number | GigabitEthernet switch-number/slot-number/port-number | Loopback interface-number | Null interface-number | Port-channel interface-number | TenGigabitEthernet switch-number/slot-number/port-number | Tunnel interface-number | Vlan interface-number }
```

<table>
<thead>
<tr>
<th>Auto-Template interface-number</th>
<th>Enables you to configure an auto-template interface. The range is from 1 to 999.</th>
</tr>
</thead>
<tbody>
<tr>
<td>GigabitEthernet switch-number/slot-number/port-number</td>
<td>Enables you to configure a Gigabit Ethernet IEEE 802.3z interface.</td>
</tr>
<tr>
<td></td>
<td>• switch-number — Switch ID. The range is from 1 to 8.</td>
</tr>
<tr>
<td></td>
<td>• slot-number</td>
</tr>
<tr>
<td></td>
<td>— Slot number. The range is from 0 to 1.</td>
</tr>
<tr>
<td></td>
<td>• port-number</td>
</tr>
<tr>
<td></td>
<td>— Port number. The range is from 1 to 48.</td>
</tr>
<tr>
<td>Loopback interface-number</td>
<td>Enables you to configure a loopback interface. The range is from 0 to 2147483647.</td>
</tr>
<tr>
<td>Null interface-number</td>
<td>Enables you to configure a null interface. The default value is 0.</td>
</tr>
<tr>
<td>Port-channel interface-number</td>
<td>Enables you to configure a port-channel interface. The range is from 1 to 128.</td>
</tr>
<tr>
<td>TenGigabitEthernet switch-number/slot-number/port-number</td>
<td>Enables you to configure a 10-Gigabit Ethernet interface.</td>
</tr>
<tr>
<td></td>
<td>• switch-number — Switch ID. The range is from 1 to 8.</td>
</tr>
<tr>
<td></td>
<td>• slot-number — Slot number. The range is from 0 to 1.</td>
</tr>
<tr>
<td></td>
<td>• port-number — Port number. The range is from 1 to 24 and 37 to 48.</td>
</tr>
<tr>
<td>Tunnel interface-number</td>
<td>Enables you to configure a tunnel interface. The range is from 0 to 2147483647.</td>
</tr>
<tr>
<td>Vlan interface-number</td>
<td>Enables you to configure a switch VLAN. The range is from 1 to 4094.</td>
</tr>
</tbody>
</table>

**Command Default** None
Command Modes

- Global configuration

Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE 3.2SE</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

This example shows how you can configure interface range:

```
Device(config)# interface range vlan 1-100
```
**ip mtu**

To set the IP maximum transmission unit (MTU) size of routed packets on all routed ports of the switch or switch stack, use the `ip mtu` command in interface configuration mode. To restore the default IP MTU size, use the `no` form of this command.

```
ip mtu  bytes
no ip mtu  bytes
```

**Syntax Description**

`bytes` MTU size, in bytes. The range is from 68 up to the system MTU value (in bytes).

**Command Default**

The default IP MTU size for frames received and sent on all switch interfaces is 1500 bytes.

**Command Modes**

Interface configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE 3.2SE</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

The upper limit of the IP value is based on the switch or switch stack configuration and refers to the currently applied system MTU value. For more information about setting the MTU sizes, see the `system mtu` global configuration command.

To return to the default IP MTU setting, you can apply the `default ip mtu` command or the `no ip mtu` command on the interface.

You can verify your setting by entering the `show ip interface interface-id` or `show interfaces interface-id` privileged EXEC command.

The following example sets the maximum IP packet size for VLAN 200 to 1000 bytes:

```
Device(config)# interface vlan 200
Device(config-if)# ip mtu 1000
```

The following example sets the maximum IP packet size for VLAN 200 to the default setting of 1500 bytes:

```
Device(config)# interface vlan 200
Device(config-if)# default ip mtu
```

This is an example of partial output from the `show ip interface interface-id` command. It displays the current IP MTU setting for the interface.

```
Device# show ip interface gigabitethernet4/0/1
GigabitEthernet4/0/1 is up, line protocol is up
Internet address is 18.0.0.1/24
Broadcast address is 255.255.255.255
Address determined by setup command
MTU is 1500 bytes
Helper address is not set
<output truncated>
```
**ipv6 mtu**

To set the IPv6 maximum transmission unit (MTU) size of routed packets on all routed ports of the switch or switch stack, use the `ipv6 mtu` command in interface configuration mode. To restore the default IPv6 MTU size, use the `no` form of this command.

```
ipv6 mtu bytes
no ipv6 mtu bytes
```

**Syntax Description**

- `bytes`: MTU size, in bytes. The range is from 1280 up to the system MTU value (in bytes).

**Command Default**

The default IPv6 MTU size for frames received and sent on all switch interfaces is 1500 bytes.

**Command Modes**

Interface configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE 3.2SE</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

The upper limit of the IPv6 MTU value is based on the switch or switch stack configuration and refers to the currently applied system MTU value. For more information about setting the MTU sizes, see the `system mtu` global configuration command.

To return to the default IPv6 MTU setting, you can apply the `default ipv6 mtu` command or the `no ipv6 mtu` command on the interface.

You can verify your setting by entering the `show ipv6 interface interface-id` or `show interface interface-id` privileged EXEC command.

The following example sets the maximum IPv6 packet size for an interface to 2000 bytes:

```
Device(config)# interface gigabitethernet4/0/1
Device(config-if)# ipv6 mtu 2000
```

The following example sets the maximum IPv6 packet size for an interface to the default setting of 1500 bytes:

```
Device(config)# interface gigabitethernet4/0/1
Device(config-if)# default ipv6 mtu
```

This is an example of partial output from the `show ipv6 interface interface-id` command. It displays the current IPv6 MTU setting for the interface.

```
Device# show ipv6 interface gigabitethernet4/0/1
GigabitEthernet4/0/1 is up, line protocol is up
   Internet address is 18.0.0.1/24
   Broadcast address is 255.255.255.255
   Address determined by setup command
   MTU is 1500 bytes
   Helper address is not set
<output truncated>
```
Ildp (interface configuration)

To enable Link Layer Discovery Protocol (LLDP) on an interface, use the `lldp` command in interface configuration mode. To disable LLDP on an interface, use the `no` form of this command.

```
lldp {med-tlv-select tlv | receive | tlv-select power-management | transmit}
nolldp {med-tlv-select tlv | receive | tlv-select power-management | transmit}
```

### Syntax Description

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>med-tlv-select</strong></td>
<td>Selects an LLDP Media Endpoint Discovery (MED) time-length-value (TLV) element to send.</td>
</tr>
<tr>
<td><strong>tlv</strong></td>
<td>String that identifies the TLV element. Valid values are the following:</td>
</tr>
<tr>
<td></td>
<td>• inventory-management — LLDP MED Inventory Management TLV.</td>
</tr>
<tr>
<td></td>
<td>• location — LLDP MED Location TLV.</td>
</tr>
<tr>
<td></td>
<td>• network-policy — LLDP MED Network Policy TLV.</td>
</tr>
<tr>
<td></td>
<td>• power-management — LLDP MED Power Management TLV.</td>
</tr>
<tr>
<td><strong>receive</strong></td>
<td>Enables the interface to receive LLDP transmissions.</td>
</tr>
<tr>
<td><strong>tlv-select</strong></td>
<td>Selects the LLDP TLVs to send.</td>
</tr>
<tr>
<td><strong>power-management</strong></td>
<td>Sends the LLDP Power Management TLV.</td>
</tr>
<tr>
<td><strong>transmit</strong></td>
<td>Enables LLDP transmission on the interface.</td>
</tr>
</tbody>
</table>

### Command Default

LLDP is disabled.

### Command Modes

Interface configuration

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE 3.2SE</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

This command is supported on 802.1 media types.

If the interface is configured as a tunnel port, LLDP is automatically disabled.

The following example shows how to disable LLDP transmission on an interface:

```
Device(config)# interface gigabitethernet1/0/1
Device(config-if)# no lldp transmit
```

The following example shows how to enable LLDP transmission on an interface:

```
Device(config)# interface gigabitethernet1/0/1
```
Device(config-if)# lldp transmit
logging event power-inline-status

To enable the logging of Power over Ethernet (PoE) events, use the logging event power-inline-status command in interface configuration mode. To disable the logging of PoE status events, use the no form of this command.

logging event power-inline-status
no logging event power-inline-status

Syntax Description

This command has no arguments or keywords.

Command Default

Logging of PoE events is enabled.

Command Modes

Interface configuration

Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE 3.2SE</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

Usage Guidelines

The no form of this command does not disable PoE error events.

Examples

This example shows how to enable logging of PoE events on a port:

Device(config-if)# interface gigabitethernet1/0/1
Device(config-if)# logging event power-inline-status
Device(config-if)#
To enable the automatic medium-dependent interface crossover (auto-MDIX) feature on the interface, use the `mdix auto` command in interface configuration mode. To disable auto-MDIX, use the `no` form of this command.

**Syntax Description**

This command has no arguments or keywords.

**Command Default**

Auto-MDIX is enabled.

**Command Modes**

Interface configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE 3.2SE</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

When auto-MDIX is enabled, the interface automatically detects the required cable connection type (straight-through or crossover) and configures the connection appropriately.

When you enable auto-MDIX on an interface, you must also set the interface speed and duplex to `auto` so that the feature operates correctly.

When auto-MDIX (and autonegotiation of speed and duplex) is enabled on one or both of the connected interfaces, link up occurs, even if the cable type (straight-through or crossover) is incorrect.

This example shows how to enable auto-MDIX on a port:

```
Device# configure terminal
Device(config)# interface gigabitethernet1/0/1
Device(config-if)# speed auto
Device(config-if)# duplex auto
Device(config-if)# mdix auto
Device(config-if)# end
```
mode (power-stack configuration)

To configure power stack mode for the power stack, use the `mode` command in power-stack configuration mode. To return to the default settings, use the `no` form of the command.

```bash
mode {power-shared | redundant} [strict]
no mode
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>power-shared</td>
<td>Sets the power stack to operate in power-shared mode. This is the default.</td>
</tr>
<tr>
<td>redundant</td>
<td>Sets the power stack to operate in redundant mode. The largest power supply</td>
</tr>
<tr>
<td></td>
<td>is removed from the power pool to be used as backup power in case one of</td>
</tr>
<tr>
<td></td>
<td>the other power supplies fails.</td>
</tr>
<tr>
<td>strict</td>
<td>(Optional) Configures the power stack mode to run a strict power budget.</td>
</tr>
<tr>
<td></td>
<td>The stack power needs cannot exceed the available power.</td>
</tr>
</tbody>
</table>

**Command Default**

The default modes are `power-shared` and non-strict.

**Command Modes**

Power-stack configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE 3.2SE</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

This command is available only on switch stacks running the IP Base or IP Services feature set.

To access power-stack configuration mode, enter the `stack-powerstack power stack name` global configuration command.

Entering the `no mode` command sets the switch to the defaults of `power-shared` and non-strict mode.

**Note**

For stack power, available power is the total power available for PoE from all power supplies in the power stack, available power is the power allocated to all powered devices connected to PoE ports in the stack, and consumed power is the actual power consumed by the powered devices.

In `power-shared` mode, all of the input power can be used for loads, and the total available power appears as one large power supply. The power budget includes all power from all supplies. No power is set aside for power supply failures. If a power supply fails, load shedding (shutting down of powered devices or switches) might occur.

In `redundant` mode, the largest power supply is removed from the power pool to use as backup power in case one of the other power supplies fails. The available power budget is the total power minus the largest power supply. This reduces the available power in the pool for switches and powered devices, but in case of a failure or an extreme power load, there is less chance of having to shut down switches or powered devices.

In `strict` mode, when a power supply fails and the available power drops below the budgeted power, the system balances the budget through load shedding of powered devices, even if the actual power is less than the available power. In nonstrict mode, the power stack can run in an over-allocated state and is stable as long as
the actual power does not exceed the available power. In this mode, a powered device drawing more than normal power could cause the power stack to start shedding loads. This is normally not a problem because most devices do not run at full power. The chances of multiple powered devices in the stack requiring maximum power at the same time is small.

In both strict and nonstrict modes, power is denied when there is no power available in the power budget.

This is an example of setting the power stack mode for the stack named power1 to power-shared with strict power budgeting. All power in the stack is shared, but when the total available power is allotted, no more devices are allowed power.

```
Device(config)# stack-power stack power1
Device(config-stackpower)# mode power-shared strict
Device(config-stackpower)# exit
```

This is an example of setting the power stack mode for the stack named power2 to redundant. The largest power supply in the stack is removed from the power pool to provide redundancy in case one of the other supplies fails.

```
Device(config)# stack-power stack power2
Device(config-stackpower)# mode redundant
Device(config-stackpower)# exit
```
network-policy

To apply a network-policy profile to an interface, use the `network-policy` command in interface configuration mode. To remove the policy, use the `no` form of this command.

```
network-policy  profile-number
no  network-policy
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>profile-number</code></td>
<td>The network-policy profile number to apply to the interface.</td>
</tr>
</tbody>
</table>

**Command Default**

No network-policy profiles are applied.

**Command Modes**

Interface configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE 3.2SE</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `network-policy  profile number` interface configuration command to apply a profile to an interface.

You cannot apply the `switchport voice vlan` command on an interface if you first configure a network-policy profile on it. However, if `switchport voice vlan  vlan-id` is already configured on the interface, you can apply a network-policy profile on the interface. The interface then has the voice or voice-signaling VLAN network-policy profile applied.

This example shows how to apply network-policy profile 60 to an interface:

```
Device(config)# interface gigabitethernet1/0/1
Device(config-if)# network-policy 60
```
network-policy profile (global configuration)

To create a network-policy profile and to enter network-policy configuration mode, use the `network-policy profile` command in global configuration mode. To delete the policy and to return to global configuration mode, use the `no` form of this command.

```
network-policy profile profile-number
no network-policy profile profile-number
```

**Syntax Description**
- `profile-number`: Network-policy profile number. The range is 1 to 4294967295.

**Command Default**
No network-policy profiles are defined.

**Command Modes**
- Global configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE 3.2SE</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `network-policy profile` global configuration command to create a profile and to enter network-policy profile configuration mode.

To return to privileged EXEC mode from the network-policy profile configuration mode, enter the `exit` command.

When you are in network-policy profile configuration mode, you can create the profile for voice and voice signaling by specifying the values for VLAN, class of service (CoS), differentiated services code point (DSCP), and tagging mode.

These profile attributes are contained in the Link Layer Discovery Protocol for Media Endpoint Devices (LLDP-MED) network-policy time-length-value (TLV).

This example shows how to create network-policy profile 60:

```
Device(config)# network-policy profile 60
Device(config-network-policy)#
```
power efficient-ethernet auto

To enable Energy Efficient Ethernet (EEE) for an interface, use the `power efficient-ethernet auto` command in interface configuration mode. To disable EEE on an interface, use the `no` form of this command.

```
power efficient-ethernet auto
no power efficient-ethernet auto
```

**Syntax Description**

This command has no arguments or keywords.

**Command Default**

EEE is disabled.

**Command Modes**

Interface configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE 3.2SE</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

You can enable EEE on devices that support low power idle (LPI) mode. Such devices can save power by entering LPI mode during periods of low utilization. In LPI mode, systems on both ends of the link can save power by shutting down certain services. EEE provides the protocol needed to transition into and out of LPI mode in a way that is transparent to upper layer protocols and applications.

The `power efficient-ethernet auto` command is available only if the interface is EEE capable. To check if an interface is EEE capable, use the `show eee capabilities` EXEC command.

When EEE is enabled, the device advertises and autonegotiates EEE to its link partner. To view the current EEE status for an interface, use the `show eee status` EXEC command.

This command does not require a license.

This example shows how to enable EEE for an interface:

```
Device(config-if)# power efficient-ethernet auto
Device(config-if)#
```

This example shows how to disable EEE for an interface:

```
Device(config-if)# no power efficient-ethernet auto
Device(config-if)#
```
To configure Cisco StackPower power-priority values for a switch in a power stack and for its high-priority and low-priority PoE ports, use the `power-priority` command in switch stack-power configuration mode. To return to the default setting, use the `no` form of the command.

```
power-priority {high  value | low  value | switch value}
no power-priority {high | low | switch}
```

**Syntax Description**

- `high value` Sets the power priority for the ports configured as high-priority ports. The range is 1 to 27, with 1 as the highest priority. The `high` value must be lower than the value set for the low-priority ports and higher than the value set for the switch.

- `low value` Sets the power priority for the ports configured as low-priority ports. The range is 1 to 27. The `low` value must be higher than the value set for the high-priority ports and the value set for the switch.

- `switch value` Sets the power priority for the switch. The range is 1 to 27. The `switch` value must be lower than the values set for the low and high-priority ports.

**Command Default**

If no values are configured, the power stack randomly determines a default priority.

The default ranges are 1 to 9 for switches, 10 to 18 for high-priority ports, 19 to 27 for low-priority ports.

On non-PoE switches, the high and low values (for port priority) have no effect.

**Command Modes**

Switch stack-power configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE 3.2SE</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

To access switch stack-power configuration mode, enter the `stack-power switch switch-number` global configuration command.

Cisco StackPower power-priority values determine the order for shutting down switches and ports when power is lost and load shedding must occur. Priority values are from 1 to 27; the highest numbers are shut down first.

We recommend that you configure different priority values for each switch and for its high priority ports and low priority ports to limit the number of devices shut down at one time during a loss of power. If you try to configure the same priority value on different switches in a power stack, the configuration is allowed, but you receive a warning message.

**Note**

This command is available only on switch stacks running the IP Base or IP Services feature set.

**Examples**

This is an example of setting the power priority for switch 1 in power stack a to 7, for the high-priority ports to 11, and for the low-priority ports to 20.
Device(config)# stack-power switch 1
Device(config-switch-stackpower)# stack-id power_stack_a
Device(config-switch-stackpower)# power-priority high 11
Device(config-switch-stackpower)# power-priority low 20
Device(config-switch-stackpower)# power-priority switch 7
Device(config-switch-stackpower)# exit
## power inline

To configure the power management mode on Power over Ethernet (PoE) ports, use the `power inline` command in interface configuration mode. To return to the default settings, use the `no` form of this command.

```
power inline {auto [max max-wattage] | never | port priority {high | low} | static [max max-wattage]}
no power inline {auto | never | port priority {high | low} | static [max max-wattage]}
```

### Syntax Description

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>auto</code></td>
<td>Enables powered-device detection. If enough power is available, automatically allocates power to the PoE port after device detection. Allocation is first-come, first-serve.</td>
</tr>
<tr>
<td><code>max max-wattage</code></td>
<td>(Optional) Limits the power allowed on the port. The range is 4000 to 30000 mW. If no value is specified, the maximum is allowed.</td>
</tr>
<tr>
<td><code>never</code></td>
<td>Disables device detection, and disables power to the port.</td>
</tr>
<tr>
<td><code>port</code></td>
<td>Configures the power priority of the port. The default priority is low.</td>
</tr>
<tr>
<td>`priority {high</td>
<td>low}`</td>
</tr>
<tr>
<td><code>static</code></td>
<td>Enables powered-device detection. Pre-allocates (reserves) power for a port before the switch discovers the powered device. This action guarantees that the device connected to the interface receives enough power.</td>
</tr>
</tbody>
</table>

### Command Default

- The default is `auto` (enabled).
- The maximum wattage is 30,000 mW.
- The default port priority is low.
### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE 3.2SE</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>Cisco IOS XE 3.3SE</td>
<td>The <strong>four-pair forced</strong> keywords were added.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

This command is supported only on PoE-capable ports. If you enter this command on a port that does not support PoE, this error message appears:

```
Device(config)# interface gigabitethernet1/0/1
Device(config-if)# power inline auto
^%
% Invalid input detected at '^' marker.
```

In a switch stack, this command is supported on all ports in the stack that support PoE.

Use the `max max-wattage` option to disallow higher-power powered devices. With this configuration, when the powered device sends Cisco Discovery Protocol (CDP) messages requesting more power than the maximum wattage, the switch removes power from the port. If the powered-device IEEE class maximum is greater than the maximum wattage, the switch does not power the device. The power is reclaimed into the global power budget.

#### Note

The switch never powers any class 0 or class 3 device if the `power inline max max-wattage` command is configured for less than 30 W.

If the switch denies power to a powered device (the powered device requests more power through CDP messages or if the IEEE class maximum is greater than the maximum wattage), the PoE port is in a power-deny state. The switch generates a system message, and the `Oper` column in the `show power inline` privileged EXEC command output shows `power-deny`.

Use the `power inline static max max-wattage` command to give a port high priority. The switch allocates PoE to a port configured in static mode before allocating power to a port configured in auto mode. The switch reserves power for the static port when it is configured rather than upon device discovery. The switch reserves the power on a static port even when there is no connected device and whether or not the port is in a shutdown or in a no shutdown state. The switch allocates the configured maximum wattage to the port, and the amount is never adjusted through the IEEE class or by CDP messages from the powered device. Because power is pre-allocated, any powered device that uses less than or equal to the maximum wattage is guaranteed power when it is connected to a static port. However, if the powered device IEEE class is greater than the maximum wattage, the switch does not supply power to it. If the switch learns through CDP messages that the powered device needs more than the maximum wattage, the powered device is shut down.

If the switch cannot pre-allocate power when a port is in static mode (for example, because the entire power budget is already allocated to other auto or static ports), this message appears: Command rejected: power inline static: pwr not available. The port configuration remains unchanged.

When you configure a port by using the `power inline auto` or the `power inline static` interface configuration command, the port autonegotiates by using the configured speed and duplex settings. This is necessary to determine the power requirements of the connected device (whether or not it is a powered device). After the power requirements have been determined, the switch hardcodes the interface by using the configured speed and duplex settings without resetting the interface.
When you configure a port by using the **power inline never** command, the port reverts to the configured speed and duplex settings.

If a port has a Cisco powered device connected to it, you should not use the **power inline never** command to configure the port. A false link-up can occur, placing the port in an error-disabled state.

Use the **power inline port priority {high | low}** command to configure the power priority of a PoE port. Powered devices connected to ports with low port priority are shut down first in case of a power shortage.

You can verify your settings by entering the **show power inline** EXEC command.

### Examples

This example shows how to enable detection of a powered device and to automatically power a PoE port on a switch:

```bash
Device(config)# interface gigabitethernet1/0/2
Device(config-if)# power inline auto
```

This example shows how to configure a PoE port on a switch to allow a class 1 or a class 2 powered device:

```bash
Device(config)# interface gigabitethernet1/0/2
Device(config-if)# power inline auto max 7000
```

This example shows how to disable powered-device detection and to not power a PoE port on a switch:

```bash
Device(config)# interface gigabitethernet1/0/2
Device(config-if)# power inline never
```

This example shows how to set the priority of a port to high, so that it would be one of the last ports to be shut down in case of power supply failure:

```bash
Device(config)# interface gigabitethernet1/0/2
Device(config-if)# power inline port priority high
```
power inline police

To enable policing of real-time power consumption on a powered device, use the `power inline police` command in interface configuration mode. To disable this feature, use the `no` form of this command.

```
power inline police [action {errdisable | log}]
no power inline police
```

**Syntax Description**

- `action` (Optional) Configures the device to turn off power to the port if the real-time power consumption exceeds the maximum power allocation on the port. This is the default action.
- `errdisable` (Optional) Configures the device to generate a syslog message while still providing power to a connected device if the real-time power consumption exceeds the maximum power allocation on the port.

**Command Default**

Policing of the real-time power consumption of the powered device is disabled.

**Command Modes**

Interface configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE 3.2SE</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

This command is supported only on the LAN Base image.

This command is supported only on Power over Ethernet (PoE)-capable ports. If you enter this command on a device or port that does not support PoE, an error message appears.

In a switch stack, this command is supported on all switches or ports in the stack that support PoE and real-time power-consumption monitoring.

When policing of the real-time power consumption is enabled, the device takes action when a powered device consumes more power than the allocated maximum amount.

When PoE is enabled, the device senses the real-time power consumption of the powered device. This feature is called `power monitoring` or `power sensing`. The device also polices the power usage with the `power policing` feature.

When power policing is enabled, the device uses one of the these values as the cutoff power on the PoE port in this order:

1. The user-defined power level that limits the power allowed on the port when you enter the `power inline auto max max-wattage` or the `power inline static max max-wattage` interface configuration command
2. The device automatically sets the power usage of the device by using CDP power negotiation or by the IEEE classification and LLDP power negotiation.

If you do not manually configure the cutoff-power value, the device automatically determines it by using CDP power negotiation or the device IEEE classification and LLDP power negotiation. If CDP or LLDP are not enabled, the default value of 30 W is applied. However without CDP or LLDP, the device does not allow devices to consume more than 15.4 W of power because values from 15400 to 30000 mW are only allocated based on CDP or LLDP requests. If a powered device consumes more than 15.4 W without CDP or LLDP
negotiation, the device might be in violation of the maximum current \( I_{max} \) limitation and might experience an \( I_{cut} \) fault for drawing more current than the maximum. The port remains in the fault state for a time before attempting to power on again. If the port continuously draws more than 15.4 W, the cycle repeats.

When a powered device connected to a PoE+ port restarts and sends a CDP or LLDP packet with a power TLV, the device locks to the power-negotiation protocol of that first packet and does not respond to power requests from the other protocol. For example, if the device is locked to CDP, it does not provide power to devices that send LLDP requests. If CDP is disabled after the device has locked on it, the device does not respond to LLDP power requests and can no longer power on any accessories. In this case, you should restart the powered device.

If power policing is enabled, the device polices power usage by comparing the real-time power consumption to the maximum power allocated on the PoE port. If the device uses more than the maximum power allocation (or cutoff power) on the port, the device either turns power off to the port, or the device generates a syslog message and updates the LEDs (the port LEDs are blinking amber) while still providing power to the device.

- To configure the device to turn off power to the port and put the port in the error-disabled state, use the `power inline police` interface configuration command.
- To configure the device to generate a syslog message while still providing power to the device, use the `power inline police action log` command.

If you do not enter the `action log` keywords, the default action is to shut down the port, turn off power to it, and put the port in the PoE error-disabled state. To configure the PoE port to automatically recover from the error-disabled state, use the `errdisable detect cause inline-power` global configuration command to enable error-disabled detection for the PoE cause and the `errdisable recovery cause inline-power interval interval` global configuration command to enable the recovery timer for the PoE error-disabled cause.

---

**Caution**

If policing is disabled, no action occurs when the powered device consumes more than the maximum power allocation on the port, which could adversely affect the device.

You can verify your settings by entering the `show power inline police` privileged EXEC command.

**Examples**

This example shows how to enable policing of the power consumption and configuring the device to generate a syslog message on the PoE port on a device:

```
Device(config)# interface gigabitethernet1/0/2
Device(config-if)# power inline police action log
```
power supply

To configure and manage the internal power supplies on a switch, use the **power supply** command in privileged EXEC mode.

```
power supply stack-member-number slot {A | B} {off | on}
```

**Syntax Description**

- **stack-member-number**
  - Stack member number for which to configure the internal power supplies. The range is 1 to 9, depending on the number of switches in the stack.
  - This parameter is available only on stacking-capable switches.

- **slot**
  - Selects the switch power supply to set.

- **A**
  - Selects the power supply in slot A.

- **B**
  - Selects the power supply in slot B.
  - **Note** Power supply slot B is the closest slot to the outer edge of the switch.

- **off**
  - Sets the switch power supply to off.

- **on**
  - Sets the switch power supply to on.

**Command Default**

The switch power supply is on.

**Command Modes**

Privileged EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE 3.2SE</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>Cisco IOS XE 3.3SE</td>
<td>The <strong>slot</strong> keyword replaced the <strong>frufep</strong> keyword.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

The **power supply** command applies to a switch or to a switch stack where all switches are the same platform.

In a switch stack with the same platform switches, you must specify the stack member before entering the **slot {A | B} off or on** keywords.

To return to the default setting, use the **power supply stack-member-number on** command.

You can verify your settings by entering the **show env power** privileged EXEC command.

**Examples**

This example shows how to set the power supply in slot A to off:

```
Device> power supply 2 slot A off
Disabling Power supply A may result in a power loss to PoE devices and/or switches ... Continue? (yes/[no]): yes
Device
```
Jun 10 04:52:54.389: %PLATFORM_ENV-6-FRU_PS_OIR: FRU Power Supply 1 powered off
Jun 10 04:52:56.717: %PLATFORM_ENV-1-FAN_NOT_PRESENT: Fan is not present

This example shows how to set the power supply in slot A to on:
Device> power supply 1 slot B on
Jun 10 04:54:39.600: %PLATFORM_ENV-6-FRU_PS_OIR: FRU Power Supply 1 powered on

This example shows the output of the show env power command:

```
Device> show env power
  SW  PID      Serial#   Status     Sys Pwr PoE Pwr Watts
       ------------------ ---------- --------------- ------- ------- -----
  1A  PWR-1RUC2-640WAC DCB1705B05B OK Good    Good  250/390
  1B  Not Present
```
show eee

To display Energy Efficient Ethernet (EEE) information for an interface, use the `show eee` command in EXEC mode.

`show eee {capabilities| status} interface interface-id`

<table>
<thead>
<tr>
<th>Syntax Description</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>capabilities</td>
<td>Displays EEE capabilities for the specified interface.</td>
</tr>
<tr>
<td>status</td>
<td>Displays EEE status information for the specified interface.</td>
</tr>
<tr>
<td>interface interface-id</td>
<td>Specifies the interface for which to display EEE capabilities or status information.</td>
</tr>
</tbody>
</table>

**Command Default**

None

**Command Modes**

User EXEC
Privileged EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE 3.2SE</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

You can enable EEE on devices that support low power idle (LPI) mode. Such devices can save power by entering LPI mode during periods of low power utilization. In LPI mode, systems on both ends of the link can save power by shutting down certain services. EEE provides the protocol needed to transition into and out of LPI mode in a way that is transparent to upper layer protocols and applications.

To check if an interface is EEE capable, use the `show eee capabilities` command. You can enable EEE on an interface that is EEE capable by using the `power efficient-ethernet auto` interface configuration command.

To view the EEE status, LPI status, and wake error count information for an interface, use the `show eee status` command.

This is an example of output from the `show eee capabilities` command on an interface where EEE is enabled:

```
Device# show eee capabilities interface gigabitethernet1/0/1
Gi1/0/1
  EEE(efficient-ethernet): yes (100-Tx and 1000T auto)
  Link Partner : yes (100-Tx and 1000T auto)
```

This is an example of output from the `show eee capabilities` command on an interface where EEE is not enabled:

```
Device# show eee capabilities interface gigabitethernet2/0/1
Gi2/0/1
  EEE(efficient-ethernet): not enabled
```
Link Partner : not enabled

This is an example of output from the `show eee status` command on an interface where EEE is enabled and operational. The table that follows describes the fields in the display.

```
Device# show eee status interface gigabitethernet1/0/4
Gi1/0/4 is up
EEE(efficient-ethernet): Operational
Rx LPI Status : Received
Tx LPI Status : Received
```

This is an example of output from the `show eee status` command on an interface where EEE operational and the ports are in low power save mode:

```
Device# show eee status interface gigabitethernet1/0/3
Gi1/0/3 is up
EEE(efficient-ethernet): Operational
Rx LPI Status : Low Power
Tx LPI Status : Low Power
Wake Error Count : 0
```

This is an example of output from the `show eee status` command on an interface where EEE is not enabled because a remote link partner is incompatible with EEE:

```
Device# show eee status interface gigabitethernet1/0/3
Gi1/0/3 is down
EEE(efficient-ethernet): Disagreed
Rx LPI Status : None
Tx LPI Status : None
Wake Error Count : 0
```

Table 1: `show eee status` Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>EEE (efficient-ethernet)</td>
<td>The EEE status for the interface. This field can have any of the following values:</td>
</tr>
<tr>
<td></td>
<td>• N/A—The port is not capable of EEE.</td>
</tr>
<tr>
<td></td>
<td>• Disabled—The port EEE is disabled.</td>
</tr>
<tr>
<td></td>
<td>• Disagreed—The port EEE is not set because a remote link partner might be incompatible with EEE; either it is not EEE capable, or its EEE setting is incompatible.</td>
</tr>
<tr>
<td></td>
<td>• Operational—The port EEE is enabled and operating.</td>
</tr>
</tbody>
</table>

If the interface speed is configured as 10 Mbps, EEE is disabled internally. When the interface speed moves back to auto, 100 Mbps or 1000 Mbps, EEE becomes active again.
The Low Power Idle (LPI) status for the link partner. These fields can have any of the following values:

- **N/A**—The port is not capable of EEE.
- **Interrupted**—The link partner is in the process of moving to low power mode.
- **Low Power**—The link partner is in low power mode.
- **None**—EEE is disabled or not capable at the link partner side.
- **Received**—The link partner is in low power mode and there is traffic activity.

If an interface is configured as half-duplex, the LPI status is None, which means the interface cannot be in low power mode until it is configured as full-duplex.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rx/Tx LPI Status</td>
<td>The number of PHY wake-up faults that have occurred. A wake-up fault can occur when EEE is enabled and the connection to the link partner is broken.</td>
</tr>
<tr>
<td>Wake Error Count</td>
<td>This information is useful for PHY debugging.</td>
</tr>
</tbody>
</table>
# show env

To display fan, temperature, and power information, use the `show env` command in EXEC mode.

```
show env {all | fan | power [{all | switch [stack-member-number]}] | stack [stack-member-number] | temperature [status]}
```

## Syntax Description

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>all</code></td>
<td>Displays the fan and temperature environmental status and the status of the internal power supplies.</td>
</tr>
<tr>
<td><code>fan</code></td>
<td>Displays the switch fan status.</td>
</tr>
<tr>
<td><code>power</code></td>
<td>Displays the internal power status of the active switch.</td>
</tr>
<tr>
<td><code>all</code></td>
<td>(Optional) Displays the status of all the internal power supplies in a standalone switch when the command is entered on the switch, or in all the stack members when the command is entered on the active switch.</td>
</tr>
<tr>
<td><code>switch</code></td>
<td>(Optional) Displays the status of the internal power supplies for each switch in the stack or for the specified switch. This keyword is available only on stacking-capable switches.</td>
</tr>
<tr>
<td><code>stack-member-number</code></td>
<td>(Optional) Number of the stack member for which to display the status of the internal power supplies or the environmental status. The range is 1 to 9.</td>
</tr>
<tr>
<td><code>stack</code></td>
<td>Displays all environmental status for each switch in the stack or for the specified switch. This keyword is available only on stacking-capable switches.</td>
</tr>
<tr>
<td><code>temperature</code></td>
<td>Displays the switch temperature status.</td>
</tr>
<tr>
<td><code>status</code></td>
<td>(Optional) Displays the switch internal temperature (not the external temperature) and the threshold values.</td>
</tr>
</tbody>
</table>

## Command Default

None

## Command Modes

- User EXEC
- Privileged EXEC

## Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE 3.2SE</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

## Usage Guidelines

Use the `show env` EXEC command to display the information for the switch being accessed—a standalone switch or the active switch. Use this command with the `stack` and `switch` keywords to display all information for the stack or for the specified stack member.
If you enter the `show env temperature status` command, the command output shows the switch temperature state and the threshold level.

You can also use the `show env temperature` command to display the switch temperature status. The command output shows the green and yellow states as `OK` and the red state as `FAULTY`. If you enter the `show env all` command, the command output is the same as the `show env temperature status` command output.

**Examples**

```
Device> show env all
Switch 1 FAN 1 is OK
Switch 1 FAN 2 is OK
Switch 1 FAN 3 is OK
FAN PS-1 is NOT PRESENT
FAN PS-2 is OK
Switch 1: SYSTEM TEMPERATURE is OK

<table>
<thead>
<tr>
<th>SW</th>
<th>PID</th>
<th>Serial#</th>
<th>Status</th>
<th>Sys Pwr</th>
<th>PoE Pwr</th>
<th>Watts</th>
</tr>
</thead>
<tbody>
<tr>
<td>1A</td>
<td>Not Present</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1B</td>
<td>FWR-C1-715WAC</td>
<td>LIT150119Z1</td>
<td>OK</td>
<td>Good</td>
<td>Good</td>
<td>715</td>
</tr>
</tbody>
</table>
```

```
Device> show env fan
Switch 1 FAN 1 is OK
Switch 1 FAN 2 is OK
Switch 1 FAN 3 is OK
FAN PS-1 is NOT PRESENT
FAN PS-2 is OK

This is an example of output from the `show env power` command:

```
Device> show env power

<table>
<thead>
<tr>
<th>SW</th>
<th>PID</th>
<th>Serial#</th>
<th>Status</th>
<th>Sys Pwr</th>
<th>PoE Pwr</th>
<th>Watts</th>
</tr>
</thead>
<tbody>
<tr>
<td>1A</td>
<td>Not Present</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1B</td>
<td>FWR-C1-715WAC</td>
<td>LIT150119Z1</td>
<td>OK</td>
<td>Good</td>
<td>Good</td>
<td>715</td>
</tr>
</tbody>
</table>
```

This is an example of output from the `show env power all` command on the active switch:

```
Device> show env power all

<table>
<thead>
<tr>
<th>SW</th>
<th>PID</th>
<th>Serial#</th>
<th>Status</th>
<th>Sys Pwr</th>
<th>PoE Pwr</th>
<th>Watts</th>
</tr>
</thead>
<tbody>
<tr>
<td>1A</td>
<td>Not Present</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1B</td>
<td>FWR-C1-715WAC</td>
<td>LIT150119Z1</td>
<td>OK</td>
<td>Good</td>
<td>Good</td>
<td>715</td>
</tr>
</tbody>
</table>
```

```
Device> show env stack
SWITCH: 1
Switch 1 FAN 1 is OK
Switch 1 FAN 2 is OK
Switch 1 FAN 3 is OK
FAN PS-1 is NOT PRESENT
FAN PS-2 is OK
Switch 1: SYSTEM TEMPERATURE is OK
Temperature Value: 28 Degree Celsius
Temperature State: GREEN
Yellow Threshold : 41 Degree Celsius
Red Threshold : 56 Degree Celsius
```
Device> **show env temperature status**
Temperature Value: 33 Degree Celsius
Temperature State: GREEN
Yellow Threshold : 65 Degree Celsius
Red Threshold : 75 Degree Celsius

### Table 2: States in the show env temperature status Command Output

<table>
<thead>
<tr>
<th>State</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Green</td>
<td>The switch temperature is in the <em>normal</em> operating range.</td>
</tr>
<tr>
<td>Yellow</td>
<td>The temperature is in the <em>warning</em> range. You should check the external temperature around the switch.</td>
</tr>
<tr>
<td>Red</td>
<td>The temperature is in the <em>critical</em> range. The switch might not run properly if the temperature is in this range.</td>
</tr>
</tbody>
</table>
show errdisable detect

To display error-disabled detection status, use the `show errdisable detect` command in EXEC mode.

**Syntax Description**

This command has no arguments or keywords.

**Command Default**

None

**Command Modes**

User EXEC

Privileged EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE 3.2SE</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

A gbic-invalid error reason refers to an invalid small form-factor pluggable (SFP) module.

The error-disable reasons in the command output are listed in alphabetical order. The mode column shows how error-disable is configured for each feature.

You can configure error-disabled detection in these modes:

- **port mode**—The entire physical port is error-disabled if a violation occurs.
- **vlan mode**—The VLAN is error-disabled if a violation occurs.
- **port/vlan mode**—The entire physical port is error-disabled on some ports and is per-VLAN error-disabled on other ports.

```
Device> show errdisable detect
ErrDisable Reason     Detection Mode
--------------------- ----------- -----
arp-inspection        Enabled port
bpduguard            Enabled vlan
channel-misconfig     Enabled port
community-limit       Enabled port
dhcp-rate-limit       Enabled port
dtp-flap              Enabled port
gbic-invalid         Enabled port
inline-power          Enabled port
invalid-policy        Enabled port
l2ptguard             Enabled port
link-flap             Enabled port
loopback              Enabled port
lsgroup               Enabled port
pagp-flap             Enabled port
pssecure-violation    Enabled port
security-violatio     Enabled port
sfp-config-mismatch   Enabled port
storm-control         Enabled port
udld                  Enabled port
```
show errdisable recovery

To display the error-disabled recovery timer information, use the `show errdisable recovery` command in EXEC mode.

**Syntax Description**

This command has no arguments or keywords.

**Command Default**

None

**Command Modes**

User EXEC

Privileged EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE 3.2SE</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

A gbic-invalid error-disable reason refers to an invalid small form-factor pluggable (SFP) module interface. Though visible in the output, the unicast-flood field is not valid.

This is an example of output from the `show errdisable recovery` command:

```
Device> show errdisable recovery
ErrDisable Reason     Timer Status
------------------------------------
udld                  Disabled
bduguard              Disabled
security-violation    Disabled
channel-misconfig     Disabled
pagp-flap             Disabled
dtp-flap              Disabled
link-flap             Enabled
l2ptguard             Disabled
psecure-violation     Disabled
gbic-invalid          Disabled
dhcp-rate-limit       Disabled
unicast-flood         Disabled
storm-control         Disabled
arp-inspection        Disabled
loopback              Disabled
Timer interval:300 seconds
Interfaces that will be enabled at the next timeout:
<table>
<thead>
<tr>
<th>Interface</th>
<th>Errdisable reason</th>
<th>Time left(sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gi1/0/2</td>
<td>link-flap</td>
<td>279</td>
</tr>
</tbody>
</table>
```
# show interfaces

To display the administrative and operational status of all interfaces or for a specified interface, use the `show interfaces` command in privileged EXEC mode.

```
show interfaces [{interface-id | vlan vlan-id}] [{accounting | capabilities [module number] | debounce | description | etherchannel | flowcontrol | pruning | stats | status [{err-disabled | inactive}] | trunk}]
```

## Syntax Description

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>interface-id</strong></td>
<td>(Optional) ID of the interface. Valid interfaces include physical ports (including type, stack member for stacking-capable switches, module, and port number) and port channels. The port channel range is 1 to 48.</td>
</tr>
<tr>
<td><strong>vlan vlan-id</strong></td>
<td>(Optional) VLAN identification. The range is 1 to 4094.</td>
</tr>
<tr>
<td><strong>accounting</strong></td>
<td>(Optional) Displays accounting information on the interface, including active protocols and input and output packets and octets. <strong>Note</strong> The display shows only packets processed in software; hardware-switched packets do not appear.</td>
</tr>
<tr>
<td><strong>capabilities</strong></td>
<td>(Optional) Displays the capabilities of all interfaces or the specified interface, including the features and options that you can configure on the interface. Though visible in the command line help, this option is not available for VLAN IDs.</td>
</tr>
<tr>
<td><strong>module number</strong></td>
<td>(Optional) Displays capabilities of all interfaces on the switch or specified stack member. The range is 1 to 9. This option is not available if you entered a specific interface ID.</td>
</tr>
<tr>
<td><strong>debounce</strong></td>
<td>(Optional) Displays port debounce timer information for an interface.</td>
</tr>
<tr>
<td><strong>description</strong></td>
<td>(Optional) Displays the administrative status and description set for an interface.</td>
</tr>
<tr>
<td><strong>etherchannel</strong></td>
<td>(Optional) Displays interface EtherChannel information.</td>
</tr>
<tr>
<td><strong>flowcontrol</strong></td>
<td>(Optional) Displays interface flow control information.</td>
</tr>
<tr>
<td><strong>mtu</strong></td>
<td>(Optional) Displays the MTU for each interface or for the specified interface.</td>
</tr>
<tr>
<td><strong>pruning</strong></td>
<td>(Optional) Displays trunk VTP pruning information for the interface.</td>
</tr>
<tr>
<td><strong>stats</strong></td>
<td>(Optional) Displays the input and output packets by switching the path for the interface.</td>
</tr>
</tbody>
</table>
status  (Optional) Displays the status of the interface. A status of unsupported in the Type field means that a non-Cisco small form-factor pluggable (SFP) module is inserted in the module slot.

err-disabled  (Optional) Displays interfaces in an error-disabled state.

inactive  (Optional) Displays interfaces in an inactive state.

trunk  (Optional) Displays interface trunk information. If you do not specify an interface, only information for active trunking ports appears.

---

**Note**

Though visible in the command-line help strings, the **crb**, **fair-queue**, **irb**, **mac-accounting**, **precedence**, **random-detect**, and **rate-limit** keywords are not supported.

---

**Command Default**

None

**Command Modes**

Privileged EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE 3.2SE</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

The **show interfaces capabilities** command with different keywords has these results:

- Use the **show interface capabilities module** *number* command to display the capabilities of all interfaces on that switch in the stack. If there is no switch with that module number in the stack, there is no output.

- Use the **show interfaces** *interface-id* **capabilities** to display the capabilities of the specified interface.

- Use the **show interfaces capabilities** (with no module number or interface ID) to display the capabilities of all interfaces in the stack.

This is an example of output from the **show interfaces** command for an interface on stack member 3:

```
Device# show interfaces gigabitethernet3/0/2
GigabitEthernet3/0/2 is down, line protocol is down (notconnect)
    Hardware is Gigabit Ethernet, address is 2037.064d.4381 (bia 2037.064d.4381)
    MTU 1500 bytes, BW 1000000 Kbit/sec, DLY 10 usec,
        reliability 255/255, txload 1/255, rxload 1/255
    Encapsulation ARPA, loopback not set
    Keepalive set (10 sec)
    Auto-duplex, Auto-speed, media type is 10/100/1000BaseTX
    input flow-control is off, output flow-control is unsupported
    ARP type: ARPA, ARP Timeout 04:00:00
    Last input never, output never, output hang never
    Last clearing of "show interface" counters never
    Input queue: 0/2000/0/0 (size/max/drops/flushes); Total output drops: 0
    Queueing strategy: fifo
    Output queue: 0/40 (size/max)
    5 minute input rate 0 bits/sec, 0 packets/sec
    5 minute output rate 0 bits/sec, 0 packets/sec
```
Device# show interfaces gigabitethernet1/0/2 capabilities

<table>
<thead>
<tr>
<th>Device# show interfaces gigabitethernet1/0/2 capabilities</th>
</tr>
</thead>
<tbody>
<tr>
<td>GigabitEthernet1/0/2</td>
</tr>
<tr>
<td>Model: UA-3850-24-CR</td>
</tr>
<tr>
<td>Type: 10/100/1000BaseTX</td>
</tr>
<tr>
<td>Speed: 10, 100, 1000, auto</td>
</tr>
<tr>
<td>Duplex: full, half, auto</td>
</tr>
<tr>
<td>Trunk encaps. type: 802.1Q</td>
</tr>
<tr>
<td>Trunk mode: on, off, desirable, nonegotiate</td>
</tr>
<tr>
<td>Channel: yes</td>
</tr>
<tr>
<td>Fast Start: yes</td>
</tr>
<tr>
<td>QoS scheduling: rx-(not configurable on per port basis),</td>
</tr>
<tr>
<td>tx-(4q3t) (3t: Two configurable values and one fixed.)</td>
</tr>
<tr>
<td>CoS rewrite: yes</td>
</tr>
<tr>
<td>ToS rewrite: yes</td>
</tr>
<tr>
<td>UDLD: yes</td>
</tr>
<tr>
<td>Inline power: no</td>
</tr>
<tr>
<td>SPAN: source/destination</td>
</tr>
<tr>
<td>PortSecure: yes</td>
</tr>
<tr>
<td>Dot1x: yes</td>
</tr>
</tbody>
</table>

This is an example of output from the `show interfaces interface description` command when the interface has been described as Connects to Marketing by using the `description` interface configuration command:

Device# show interfaces gigabitethernet1/0/2 description

<table>
<thead>
<tr>
<th>Interface</th>
<th>Status</th>
<th>Protocol Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gi1/0/2</td>
<td>up</td>
<td>down Connects to Marketing</td>
</tr>
</tbody>
</table>

This is an example of output from the `show interfaces interface-id pruning` command when pruning is enabled in the VTP domain:

Device# show interfaces gigabitethernet1/0/2 pruning

<table>
<thead>
<tr>
<th>Port</th>
<th>Vlans pruned for lack of request by neighbor</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gi1/0/2</td>
<td>3, 4</td>
</tr>
</tbody>
</table>

Port Vlans traffic requested of neighbor

| Gi1/0/2 | 1-3                        |

This is an example of output from the `show interfaces stats` command for a specified VLAN interface:

Device# show interfaces vlan 1 stats

<table>
<thead>
<tr>
<th>Switching path</th>
<th>Pkts In</th>
<th>Chars In</th>
<th>Pkts Out</th>
<th>Chars Out</th>
</tr>
</thead>
<tbody>
<tr>
<td>Processor</td>
<td>1165354</td>
<td>136205310</td>
<td>570800</td>
<td>91731594</td>
</tr>
<tr>
<td>Route cache</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Total</td>
<td>1165354</td>
<td>136205310</td>
<td>570800</td>
<td>91731594</td>
</tr>
</tbody>
</table>

This is an example of output from the `show interfaces interface-id status` command:
Device# **show interfaces gigabitethernet1/0/20 status**

<table>
<thead>
<tr>
<th>Port</th>
<th>Name</th>
<th>Status</th>
<th>Vlan</th>
<th>Duplex</th>
<th>Speed</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gi1/0/20</td>
<td></td>
<td>notconnect</td>
<td>1</td>
<td>auto</td>
<td>auto</td>
<td>10/100/1000BaseTX</td>
</tr>
</tbody>
</table>

This is an example of output from the **show interfaces status err-disabled** command. It displays the status of interfaces in the error-disabled state:

Device# **show interfaces status err-disabled**

<table>
<thead>
<tr>
<th>Port</th>
<th>Name</th>
<th>Status</th>
<th>Reason</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gi1/0/2</td>
<td></td>
<td>err-disabled</td>
<td>gbic-invalid</td>
</tr>
<tr>
<td>Gi2/0/3</td>
<td></td>
<td>err-disabled</td>
<td>dtp-flap</td>
</tr>
</tbody>
</table>

This is an example of output from the **show interfaces interface-id pruning** command:

Device# **show interfaces gigabitethernet1/0/2 pruning**

Port Vlans pruned for lack of request by neighbor

Device# **show interfaces gigabitethernet1/0/1 trunk**

<table>
<thead>
<tr>
<th>Port</th>
<th>Mode</th>
<th>Encapsulation</th>
<th>Status</th>
<th>Native vlan</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gi1/0/1</td>
<td>on</td>
<td>802.1q</td>
<td>other</td>
<td>10</td>
</tr>
</tbody>
</table>

Port
Gi1/0/1 | none |

Port
Gi1/0/1 | Vlans allowed and active in management domain |

Port
Gi1/0/1 | Vlans in spanning tree forwarding state and not pruned |

Port
Gi1/0/1 | none |
show interfaces counters

To display various counters for the switch or for a specific interface, use the `show interfaces counters` command in privileged EXEC mode.

```
show interfaces [interface-id] counters [errors | etherchannel | module stack-member-number | protocol status | trunk]
```

**Syntax Description**

- **interface-id** (Optional) ID of the physical interface, including type, stack member (stacking-capable switches only) module, and port number.
- **errors** (Optional) Displays error counters.
- **etherchannel** (Optional) Displays EtherChannel counters, including octets, broadcast packets, multicast packets, and unicast packets received and sent.
- **module stack-member-number** (Optional) Displays counters for the specified stack member. The range is 1 to 9.

**Note**

In this command, the `module` keyword refers to the stack member number. The module number that is part of the interface ID is always zero.

- **protocol status** (Optional) Displays the status of protocols enabled on interfaces.
- **trunk** (Optional) Displays trunk counters.

**Note**

Though visible in the command-line help string, the `vlan vlan-id` keyword is not supported.

**Command Default**

None

**Command Modes**

Privileged EXEC

**Command History**

- **Release**
  - Cisco IOS XE 3.2SE

**Modification**

- This command was introduced.

**Usage Guidelines**

If you do not enter any keywords, all counters for all interfaces are included.

This is an example of partial output from the `show interfaces counters` command. It displays all counters for the switch.

```
Device# show interfaces counters
Port InOctets InUcastPkts InMcastPkts InBcastPkts
G11/0/1 0 0 0 0
G11/0/2 0 0 0 0
G11/0/3 95285341 43115 1178430 1950
G11/0/4 0 0 0 0
```
This is an example of partial output from the `show interfaces counters module` command for stack member 2. It displays all counters for the specified switch in the stack.

```
Device# show interfaces counters module 2
Port  InOctets  InUcastPkts  InMcastPkts  InBcastPkts
Gi1/0/1  520       2          0           0
Gi1/0/2  520       2          0           0
Gi1/0/3  520       2          0           0
Gi1/0/4  520       2          0           0
```

This is an example of partial output from the `show interfaces counters protocol status` command for all interfaces:

```
Device# show interfaces counters protocol status
Protocols allocated:
Vlan1: Other, IP
Vlan20: Other, IP, ARP
Vlan30: Other, IP, ARP
Vlan40: Other, IP, ARP
Vlan50: Other, IP, ARP
Vlan60: Other, IP, ARP
Vlan70: Other, IP, ARP
Vlan80: Other, IP, ARP
Vlan90: Other, IP, ARP
Vlan900: Other, IP, ARP
Vlan3000: Other, IP
Vlan3500: Other, IP
GigabitEthernet1/0/1: Other, IP, ARP, CDP
GigabitEthernet1/0/2: Other, IP
GigabitEthernet1/0/3: Other, IP
GigabitEthernet1/0/4: Other, IP
GigabitEthernet1/0/5: Other, IP
GigabitEthernet1/0/6: Other, IP
GigabitEthernet1/0/7: Other, IP
GigabitEthernet1/0/8: Other, IP
GigabitEthernet1/0/9: Other, IP
GigabitEthernet1/0/10: Other, IP, CDP
```

This is an example of output from the `show interfaces counters trunk` command. It displays trunk counters for all interfaces.

```
Device# show interfaces counters trunk
Port  TrunkFramesTx  TrunkFramesRx  WrongEncap
Gi1/0/1  0            0             0
Gi1/0/2  0            0             0
Gi1/0/3  80678        0             0
Gi1/0/4  82320        0             0
Gi1/0/5  0            0             0
```

<output truncated>
# show interfaces switchport

To display the administrative and operational status of a switching (nonrouting) port, including port blocking and port protection settings, use the `show interfaces switchport` command in privileged EXEC mode.

```
show interfaces [interface-id] switchport [{module number}]
```

## Syntax Description

- **interface-id**: (Optional) ID of the interface. Valid interfaces include physical ports (including type, stack member for stacking-capable switches, module, and port number) and port channels. The port channel range is 1 to 48.

- **module number**: (Optional) Displays switchport configuration of all interfaces on the switch or specified stack member.
  - The range is 1 to 9.
  - This option is not available if you entered a specific interface ID.

## Command Default

None

## Command Modes

Privileged EXEC

## Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE 3.2SE</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

## Usage Guidelines

Use the `show interface switchport module number` command to display the switch port characteristics of all interfaces on that switch in the stack. If there is no switch with that module number in the stack, there is no output.

This is an example of output from the `show interfaces switchport` command for a port. The table that follows describes the fields in the display.

> Private VLANs are not supported in this release, so those fields are not applicable.

```bash
Device# show interfaces gigabitethernet1/0/1 switchport
Name: Gi1/0/1
Switchport: Enabled
Administrative Mode: trunk
Operational Mode: down
Administrative Trunking Encapsulation: dot1q
Negotiation of Trunking: On
Access Mode VLAN: 1 (default)
Trunking Native Mode VLAN: 10 (VLAN0010)
Administrative Native VLAN tagging: enabled
Voice VLAN: none
Administrative private-vlan host-association: none
Administrative private-vlan mapping: none
Administrative private-vlan trunk native VLAN: none
Administrative private-vlan trunk Native VLAN tagging: enabled
Administrative private-vlan trunk encapsulation: dot1q
```
Administrative private-vlan trunk normal VLANs: none
Administrative private-vlan trunk associations: none
Administrative private-vlan trunk mappings: none
Operational private-vlan: none
Trunking VLANs Enabled: 11-20
Pruning VLANs Enabled: 2-1001
Capture Mode Disabled
Capture VLANs Allowed: ALL

Protected: false
Unknown unicast blocked: disabled
Unknown multicast blocked: disabled
Appliance trust: none

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Displays the port name.</td>
</tr>
<tr>
<td>Switchport</td>
<td>Displays the administrative and operational status of the port. In this display, the port is in switchport mode.</td>
</tr>
<tr>
<td>Administrative Mode</td>
<td>Displays the administrative and operational modes.</td>
</tr>
<tr>
<td>Operational Mode</td>
<td>Displays the administrative and operational modes.</td>
</tr>
<tr>
<td>Administrative Trunking Encapsulation</td>
<td>Displays the administrative and operational encapsulation method and whether trunking negotiation is enabled.</td>
</tr>
<tr>
<td>Operational Trunking Encapsulation</td>
<td></td>
</tr>
<tr>
<td>Negotiation of Trunking</td>
<td>Displays the VLAN ID to which the port is configured.</td>
</tr>
<tr>
<td>Access Mode VLAN</td>
<td>Displays the VLAN ID to which the port is configured.</td>
</tr>
<tr>
<td>Trunking Native Mode VLAN</td>
<td>Lists the VLAN ID of the trunk that is in native mode.</td>
</tr>
<tr>
<td>Trunking VLANs Enabled</td>
<td>Lists the allowed VLANs on the trunk. Lists the active VLANs on the trunk.</td>
</tr>
<tr>
<td>Trunking VLANs Active</td>
<td>Lists the VLANs that are pruning-eligible.</td>
</tr>
<tr>
<td>Pruning VLANs Enabled</td>
<td>Displays whether or not protected port is enabled (True) or disabled (False) on the interface.</td>
</tr>
<tr>
<td>Protected</td>
<td>Displays whether or not unknown multicast and unknown unicast traffic is blocked on the interface.</td>
</tr>
<tr>
<td>Unknown unicast blocked</td>
<td>Displays the class of service (CoS) setting of the data packets of the IP phone.</td>
</tr>
<tr>
<td>Unknown multicast blocked</td>
<td>Displays the VLAN ID on which voice VLAN is enabled.</td>
</tr>
<tr>
<td>Voice VLAN</td>
<td>Displays the VLAN ID on which voice VLAN is enabled.</td>
</tr>
<tr>
<td>Appliance trust</td>
<td>Displays the VLAN ID on which voice VLAN is enabled.</td>
</tr>
</tbody>
</table>
show interfaces transceiver

To display the physical properties of a small form-factor pluggable (SFP) module interface, use the `show interfaces transceiver` command in EXEC mode.

```
show interfaces [interface-id] transceiver [{detail | module number | properties | supported-list | threshold-table}]
```

**Syntax Description**

- `interface-id` (Optional) ID of the physical interface, including type, stack member (stacking-capable switches only) module, and port number.
- `detail` (Optional) Displays calibration properties, including high and low numbers and any alarm information for any Digital Optical Monitoring (DoM)-capable transceiver if one is installed in the switch.
- `module number` (Optional) Limits display to interfaces on module on the switch.
  - The range is 1 to 9.
  - This option is not available if you entered a specific interface ID.
- `properties` (Optional) Displays speed, duplex, and inline power settings on an interface.
- `supported-list` (Optional) Lists all supported transceivers.
- `threshold-table` (Optional) Displays alarm and warning threshold table.

**Command Modes**

- User EXEC
- Privileged EXEC

**Command History**

- **Release**: Cisco IOS XE 3.2SE
- **Modification**: This command was introduced.

**Examples**

This is an example of output from the `show interfaces interface-id transceiver properties` command:

```
Device# show interfaces transceiver

If device is externally calibrated, only calibrated values are printed.
NA or N/A: not applicable, Tx: transmit, Rx: receive.
mA: milliamperes, dBm: decibels (milliwatts).

<table>
<thead>
<tr>
<th>Port</th>
<th>Temperature (Celsius)</th>
<th>Voltage (Volts)</th>
<th>Current (mA)</th>
<th>Optical Tx Power (dBm)</th>
<th>Optical Rx Power (dBm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gi5/1/2</td>
<td>42.9</td>
<td>3.28</td>
<td>22.1</td>
<td>-5.4</td>
<td>-8.1</td>
</tr>
<tr>
<td>Te5/1/3</td>
<td>32.0</td>
<td>3.28</td>
<td>19.8</td>
<td>2.4</td>
<td>-4.2</td>
</tr>
</tbody>
</table>
```

```
Device# show interfaces gigabitethernet1/1 transceiver properties
```
This is an example of output from the `show interfaces interface-id transceiver detail` command:

Device# `show interfaces gigabitethernet1/1/1 transceiver detail`
ITU Channel not available (Wavelength not available),
Transceiver is internally calibrated.
mA:milliamperes, dBm:decibels (milliwatts), N/A:not applicable.
++:high alarm, +:high warning, -:low warning, -- :low alarm.
A2D readouts (if they differ), are reported in parentheses.
The threshold values are uncalibrated.

<table>
<thead>
<tr>
<th>Port</th>
<th>Temperature (Celsius)</th>
<th>High Alarm Threshold (Celsius)</th>
<th>High Warn Threshold (Celsius)</th>
<th>Low Warn Threshold (Celsius)</th>
<th>Low Alarm Threshold (Celsius)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gi1/1/1</td>
<td>29.9</td>
<td>74.0</td>
<td>70.0</td>
<td>0.0</td>
<td>-4.0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Port</th>
<th>Voltage (Volts)</th>
<th>High Alarm Threshold (Volts)</th>
<th>High Warn Threshold (Volts)</th>
<th>Low Warn Threshold (Volts)</th>
<th>Low Alarm Threshold (Volts)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gi1/1/1</td>
<td>3.28</td>
<td>3.60</td>
<td>3.50</td>
<td>3.10</td>
<td>3.00</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Port</th>
<th>Optical Transmit Power (dBm)</th>
<th>High Alarm Threshold (dBm)</th>
<th>High Warn Threshold (dBm)</th>
<th>Low Warn Threshold (dBm)</th>
<th>Low Alarm Threshold (dBm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gi1/1/1</td>
<td>1.8</td>
<td>7.9</td>
<td>3.9</td>
<td>0.0</td>
<td>-4.0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Port</th>
<th>Optical Receive Power (dBm)</th>
<th>High Alarm Threshold (dBm)</th>
<th>High Warn Threshold (dBm)</th>
<th>Low Warn Threshold (dBm)</th>
<th>Low Alarm Threshold (dBm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gi1/1/1</td>
<td>-23.5</td>
<td>-5.0</td>
<td>-9.0</td>
<td>-28.2</td>
<td>-32.2</td>
</tr>
</tbody>
</table>

Device# `show interfaces transceiver supported-list`
Transceiver Type | Cisco p/n min version supporting DOM
------------------|----------------------------------------------------------
DWDM GBIC         | ALL                                                      
DWDM SFP          | ALL                                                      
RX only WDM GBIC  | ALL                                                      
DWDM XENPAK       | ALL                                                      
DWDM X2           | ALL                                                      
DWDM XFP          | ALL                                                      
CWDM GBIC         | NONE                                                     
CWDM X2           | ALL                                                      
CWDM XFP          | ALL                                                      
XENPAK ZR         | ALL                                                      
X2 ZR             | ALL                                                      
XFP ZR            | ALL                                                      
Rx_only_WDM_XENPAK| ALL                                                      
XENPAK_ER         | 10-1888-04                                               
X2_ER             | ALL                                                      

Interface and Hardware Commands

56
This is an example of output from the `show interfaces transceiver threshold-table` command:

```
Device# show interfaces transceiver threshold-table

<table>
<thead>
<tr>
<th>Optical Tx</th>
<th>Optical Rx</th>
<th>Temp</th>
<th>Laser Bias</th>
<th>Voltage</th>
</tr>
</thead>
<tbody>
<tr>
<td>DWDM GBIC</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
HW         |            |      |            |         |
Min1       | -4.00      | -32.00| -4         | N/A     | 4.65   |
Min2       | 0.00       | -28.00| 0          | N/A     | 4.75   |
Max2       | 4.00       | -9.00 | 70         | N/A     | 5.25   |
```
### Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>transceiver type all</td>
<td>Enters the transceiver type configuration mode.</td>
</tr>
<tr>
<td>monitoring</td>
<td>Enables digital optical monitoring.</td>
</tr>
</tbody>
</table>
show memory platform

To display memory statistics of a platform, use the `show memory platform` command in privileged EXEC mode.

```
show memory platform [{compressed-swap | information | page-merging}]
```

**Syntax Description**

- `compressed-swap` (Optional) Displays platform memory compressed-swap information.
- `information` (Optional) Displays general information about the platform.
- `page-merging` (Optional) Displays platform memory page-merging information.

**Command Modes**

Privileged EXEC (#)

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE Denali 16.1.1</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Prior to Cisco IOS XE Denali 16.3.1, the Free Memory displayed in the command output was obtained from the underlying Linux kernel. This value was not accurate because some memory chunks that was available for use was not considered as free memory.

In Cisco IOS XE Denali 16.3.1, the free memory is accurately computed and displayed in the Free Memory field of the command output.

**Examples**

The following is sample output from the `show memory platform` command:

```
Switch# show memory platform

Virtual memory : 12874653696
Pages resident : 627041
Major page faults: 2220
Minor page faults: 2348631

Architecture : mips64
Memory (kB)
  Physical : 3976852
  Total : 3976852
  Used : 2761276
  Free : 1215576
  Active : 2128196
  Inactive : 1581856
  Inact-dirty : 0
  Inact-clean : 0
  Dirty : 0
  AnonPages : 1294984
  Bounce : 0
  Cached : 1978168
  Commit Limit : 1988424
  Committed As : 3343324
  High Total : 0
  High Free : 0
```
Low Total : 3976852
Low Free : 1215628
Mapped : 516212
NFS Unstable : 0
Page Tables : 17124
Slab : 0
VMmalloc Chunk : 1069542588
VMmalloc Total : 1069547512
VMmalloc Used : 2588
Writeback : 0
HugePages Total: 0
HugePages Free : 0
HugePages Rsvd : 0
HugePage Size : 2048

Swap (kB)
Total : 0
Used : 0
Free : 0
Cached : 0

Buffers (kB) : 437136

Load Average
1-Min : 1.04
5-Min : 1.16
15-Min : 0.94

The following is sample output from the show memory platform information command:

Device# show memory platform information

Virtual memory : 12870438912
Pages resident : 626833
Major page faults: 2222
Minor page faults: 2362455

Architecture : mips64

Memory (kB)
Physical : 3976852
Total : 3976852
Used : 2761224
Free : 1215628
Active : 2128060
Inactive : 1584444
Inact-dirty : 0
Inact-clean : 0
Dirty : 284
AnonPages : 1294656
Bounce : 0
Cached : 1979644
Commit Limit : 1988424
Committed As : 3342184
High Total : 0
High Free : 0
Low Total : 3976852
Low Free : 1215628
Mapped : 516212
NFS Unstable : 0
Page Tables : 17096
Slab : 0
VMmalloc Chunk : 1069542588
VMmalloc Total : 1069547512
VMmalloc Used : 2588
Writeback : 0
HugePages Total: 0
HugePages Free : 0
HugePages Rsvd : 0
HugePage Size : 2048

Swap (kB)
Total : 0
Used : 0
Free : 0
Cached : 0

Buffers (kB) : 438228

Load Average
1-Min : 1.54
5-Min : 1.27
15-Min : 0.99
show module

To display module information such as switch number, model number, serial number, hardware revision number, software version, MAC address and so on, use this command in user EXEC or privileged EXEC mode.

```
show module [{switch-num}]
```

<table>
<thead>
<tr>
<th>Syntax Description</th>
<th>switch-num</th>
<th>(Optional) Number of the switch.</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Command Default</th>
<th>None</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Command Modes</th>
<th>User EXEC (&gt;)</th>
<th>Privileged EXEC (#)</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Command History</th>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Cisco IOS XE Denali 16.1.1</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Usage Guidelines</th>
<th>Entering the show module command without the switch-num argument is the same as entering the show module all command.</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Examples</th>
<th>This example shows how to display information for all the modules on a Cisco Catalyst 3850 Series switch:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Device# show module</td>
<td>Device# show module</td>
</tr>
<tr>
<td>Switch   Ports Model Serial No. MAC address Hw Ver. Sw Ver.</td>
<td>Switch   Ports Model Serial No. MAC address Hw Ver. Sw Ver.</td>
</tr>
</tbody>
</table>
| ------    ------  ----------- -------------- --------------  -------  ----------- 
| 2        32      WS-C3850-24P FCW1934D0ZN 3820.56b8.d000 V06 16.01.00xxx |
**show mgmt-infra trace messages ilpower**

To display inline power messages within a trace buffer, use the `show mgmt-infra trace messages ilpower` command in privileged EXEC mode.

**Syntax Description**

```
show mgmt-infra trace messages ilpower [switch stack-member-number]
```

- `switch stack-member-number` (Optional) Specifies the stack member number for which to display inline power messages within a trace buffer.

**Command Default**

None

**Command Modes**

Privileged EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE 3.2SE</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

This is an output example from the `show mgmt-infra trace messages ilpower` command:

```
Device# show mgmt-infra trace messages ilpower
[10/23/12 14:05:10.984 UTC 1 3] Initialized inline power system configuration for slot 1.
[10/23/12 14:05:10.984 UTC 2 3] Initialized inline power system configuration for slot 2.
[10/23/12 14:05:10.984 UTC 3 3] Initialized inline power system configuration for slot 3.
[10/23/12 14:05:10.984 UTC 5 3] Initialized inline power system configuration for slot 5.
[10/23/12 14:05:10.984 UTC 7 3] Initialized inline power system configuration for slot 7.
[10/23/12 14:05:10.984 UTC 8 3] Initialized inline power system configuration for slot 8.
[10/23/12 14:05:10.984 UTC a 3] Inline power subsystem initialized.
[10/23/12 14:05:18.908 UTC b 264] Create new power pool for slot 1
[10/23/12 14:05:18.909 UTC c 264] Set total inline power to 450 for slot 1
[10/23/12 14:05:20.273 UTC d 3] PoE is not supported on.
[10/23/12 14:05:20.288 UTC e 3] PoE is not supported on.
[10/23/12 14:05:20.299 UTC f 3] PoE is not supported on.
[10/23/12 14:05:20.311 UTC 10 3] PoE is not supported on.
[10/23/12 14:05:20.373 UTC 11 98] Inline power process post for switch 1
[10/23/12 14:05:20.373 UTC 12 98] PoE post passed on switch 1
[10/23/12 14:05:20.379 UTC 13 3] Slot #1: PoE initialization for board id 16387
[10/23/12 14:05:20.379 UTC 14 3] Set total inline power to 450 for slot 1
[10/23/12 14:05:20.379 UTC 15 3] Gi1/0/1 port config Initialized
[10/23/12 14:05:20.380 UTC 17 3] Gi1/0/24 port config Initialized
```
show mgmt-infra trace messages ilpower

[10/23/12 14:05:50.440 UTC 1a 3] Slot #1: PoE initialization for board id 16387
[10/23/12 14:05:50.440 UTC 1b 3] Duplicate init event
show mgmt-infra trace messages ilpower-ha

To display inline power high availability messages within a trace buffer, use the show mgmt-infra trace messages ilpower-ha command in privileged EXEC mode.

```
switch stack-member-number
```

(Optional) Specifies the stack member number for which to display inline power messages within a trace buffer.

---

### Syntax Description

<table>
<thead>
<tr>
<th>Command Mode</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>Privileged EXEC</td>
<td>None</td>
</tr>
</tbody>
</table>

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE 3.2SE</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

This is an output example from the show mgmt-infra trace messages ilpower-ha command:

```
Device# show mgmt-infra trace messages ilpower-ha
```
show mgmt-infra trace messages platform-mgr-poe

To display platform manager Power over Ethernet (PoE) messages within a trace buffer, use the show mgmt-infra trace messages platform-mgr-poe privileged EXEC command.

show mgmt-infra trace messages platform-mgr-poe [switch stack-member-number]

Syntax Description

switch stack-member-number (Optional) Specifies the stack member number for which to display messages within a trace buffer.

Command Default
None

Command Modes
Privileged EXEC

Command History

Release Modification
Cisco IOS XE 3.2SE This command was introduced.

This is an example of partial output from the show mgmt-infra trace messages platform-mgr-poe command:

Device# show mgmt-infra trace messages platform-mgr-poe
[10/23/12 14:04:06.431 UTC 1 5495] PoE Info: get power controller param sent:
[10/23/12 14:04:06.431 UTC 2 5495] PoE Info: POE_SHUT sent for port 1 (0:0)
[10/23/12 14:04:06.431 UTC 3 5495] PoE Info: POE_SHUT sent for port 2 (0:1)
[10/23/12 14:04:06.431 UTC 4 5495] PoE Info: POE_SHUT sent for port 3 (0:2)
[10/23/12 14:04:06.431 UTC 5 5495] PoE Info: POE_SHUT sent for port 4 (0:3)
[10/23/12 14:04:06.431 UTC 6 5495] PoE Info: POE_SHUT sent for port 5 (0:4)
[10/23/12 14:04:06.431 UTC 7 5495] PoE Info: POE_SHUT sent for port 6 (0:5)
[10/23/12 14:04:06.431 UTC 8 5495] PoE Info: POE_SHUT sent for port 7 (0:6)
[10/23/12 14:04:06.431 UTC 9 5495] PoE Info: POE_SHUT sent for port 8 (0:7)
[10/23/12 14:04:06.431 UTC a 5495] PoE Info: POE_SHUT sent for port 9 (0:8)
[10/23/12 14:04:06.431 UTC b 5495] PoE Info: POE_SHUT sent for port 10 (0:9)
[10/23/12 14:04:06.431 UTC c 5495] PoE Info: POE_SHUT sent for port 11 (0:10)
[10/23/12 14:04:06.431 UTC d 5495] PoE Info: POE_SHUT sent for port 12 (0:11)
[10/23/12 14:04:06.431 UTC e 5495] PoE Info: POE_SHUT sent for port 13 (e:0)
[10/23/12 14:04:06.431 UTC f 5495] PoE Info: POE_SHUT sent for port 14 (e:1)
[10/23/12 14:04:06.431 UTC 10 5495] PoE Info: POE_SHUT sent for port 15 (e:2)
[10/23/12 14:04:06.431 UTC 11 5495] PoE Info: POE_SHUT sent for port 16 (e:3)
[10/23/12 14:04:06.431 UTC 12 5495] PoE Info: POE_SHUT sent for port 17 (e:4)
[10/23/12 14:04:06.431 UTC 13 5495] PoE Info: POE_SHUT sent for port 18 (e:5)
[10/23/12 14:04:06.431 UTC 14 5495] PoE Info: POE_SHUT sent for port 19 (e:6)
[10/23/12 14:04:06.431 UTC 15 5495] PoE Info: POE_SHUT sent for port 20 (e:7)
[10/23/12 14:04:06.431 UTC 16 5495] PoE Info: POE_SHUT sent for port 21 (e:8)
[10/23/12 14:04:06.431 UTC 17 5495] PoE Info: POE_SHUT sent for port 22 (e:9)
[10/23/12 14:04:06.431 UTC 18 5495] PoE Info: POE_SHUT sent for port 23 (e:10)
show network-policy profile

To display the network-policy profiles, use the `show network policy profile` command in privileged EXEC mode.

```
show network-policy profile [profile-number]
```

**Syntax Description**

| profile-number | (Optional) Displays the network-policy profile number. If no profile is entered, all network-policy profiles appear. |

**Command Default**

None

**Command Modes**

Privileged EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE 3.2SE</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

This is an example of output from the `show network-policy profile` command:

```
Device# show network-policy profile
Network Policy Profile 60
    Interface:
    none
```
show platform hardware fed switch forward

To display device-specific hardware information, use the `show platform hardware fed switch switch_number` command.

This topic elaborates only the forwarding-specific options, that is, the options available with the `show platform hardware fed switch {switch_num | active | standby} forward summary` command.

The output of the `show platform hardware fed switch switch_number forward summary` displays all the details about the forwarding decision taken for the packet.

```
show platform hardware fed switch {switch_num | active | standby} forward summary
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>switch</td>
<td>The switch for which you want to display information. You have the following options:</td>
</tr>
<tr>
<td></td>
<td>• <code>switch_num</code> — ID of the switch.</td>
</tr>
<tr>
<td></td>
<td>• <code>active</code> — Displays information relating to the active switch.</td>
</tr>
<tr>
<td></td>
<td>• <code>standby</code> — Displays information relating to the standby switch, if available.</td>
</tr>
<tr>
<td>forward summary</td>
<td>Displays packet forwarding information.</td>
</tr>
</tbody>
</table>

**Note**
The keyword `summary` was added in the Cisco IOS XE Denali 16.3.1 release.

Support for the keyword `summary` has been discontinued in the Cisco IOS XE Everest 16.6.1 release and later releases.

**Command Modes**

Privileged EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE 3.2SE</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>Cisco IOS XE Denali 16.3.1</td>
<td>Support for the keyword <code>summary</code> was added.</td>
</tr>
<tr>
<td>Cisco IOS XE Everest 16.6.1 and later releases</td>
<td>Support for the keyword <code>summary</code> was discontinued.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

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

Fields displayed in the command output are explained below.

- **Station Index**: The Station Index is the result of the layer 2 lookup and points to a station descriptor which provides the following:
  - **Destination Index**: Determines the egress port(s) to which the packets should be sent to. Global Port Number(GPN) can be used as the destination index. A destination index with 15 down to 12
bits set indicates the GPN to be used. For example, destination index - 0xF04E corresponds to GPN - 78 (0x4e).

- **Rewrite Index**: Determines what needs to be done with the packets. For layer 2 switching, this is typically a bridging action.

- **Flexible Lookup Pipeline Stages (FPS)**: Indicates the forwarding decision that was taken for the packet - routing or bridging.

- **Replication Bit Map**: Determines if the packets should be sent to CPU or stack.
  - Local Data Copy = 1
  - Remote Data copy = 0
  - Local CPU Copy = 0
  - Remote CPU Copy = 0

**Example**

This is an example of output from the `show platform hardware fed switch {switch_num | active | standby} forward summary` command.

Device# `show platform hardware fed switch 1 forward summary`
Time: Fri Sep 16 08:25:00 PDT 2016

Incomming Packet Details:

```
###[ Ethernet ]###
dst = 00:51:0f:f2:0e:11
src = 00:1d:01:85:ba:22
type = ARP
###[ ARP ]###
hwtype = 0x1
ptype = IPv4
hwlen = 6
plen = 4
op = is-at
hwsrc = 00:1d:01:85:ba:22
psrc = 10.10.1.33
hwdst = 00:51:0f:f2:0e:11
pdst = 10.10.1.1
```

Ingress:
Switch : 1
Port : GigabitEthernet1/0/1
Global Port Number : 1
Local Port Number : 1
Asic Port Number : 21
ASIC Number : 0
STP state :
  blkLrn31to0: 0xffdfffdf
  blkFwd31to0: 0xffdfffdf
Vlan : 1
Station Descriptor : 170
DestIndex : 0xF009
DestModIndex : 2
RewriteIndex : 2
Forwarding Decision: FPS 2A L2 Destination
Replication Bitmap:
Local CPU copy : 0
Local Data copy : 1
Remote CPU copy : 0
Remote Data copy : 0

Egress:
Switch : 1
Outgoing Port : GigabitEthernet1/0/9
Global Port Number : 9
ASIC Number : 0
Vlan : 1
show platform hardware fed switch forward interface

To debug forwarding information and to trace the packet path in the hardware forwarding plane, use the `show platform hardware fed switch switch_number forward interface` command. This command simulates a user-defined packet and retrieves the forwarding information from the hardware forwarding plane. A packet is generated on the ingress port based on the packet parameters that you have specified in this command. You can also provide a complete packet from the captured packets stored in a PCAP file.

This topic elaborates only the interface forwarding-specific options, that is, the options available with the `show platform hardware fed switch {switch_num | active | standby} forward interface` command.

```
show platform hardware fed switch {switch_num | active | standby} forward interface interface-type interface-number source-mac-address destination-mac-address {protocol-number | arp | cos | ipv4 | ipv6 | mpls}
show platform hardware fed switch {switch_num | active | standby} forward interface interface-type interface-number pcap pcap-file-name number packet-number data
show platform hardware fed switch {switch_num | active | standby} forward interface interface-type interface-number vlan vlan-id source-mac-address destination-mac-address {protocol-number | arp | cos | ipv4 | ipv6 | mpls}
```

### Syntax Description

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
</table>
| switch {switch_num | active | standby} | The switch on which packet tracing has to be scheduled. The input port should be available on this switch. You have the following options:
| interface interface-type interface-number | The input interface on which packet trace is simulated. |
| source-mac-address | The source MAC address of the packet you want to simulate. |
| destination-mac-address | The MAC address of the destination interface in hexadecimal format. |
| protocol-number | The number assigned to any L3 protocol. |
| arp | The Address Resolution Protocol (ARP) parameters. |
| ipv4 | The IPv4 packet parameters. |
| ipv6 | The IPv6 packet parameters. |
| mpls | The Multiprotocol Label Switching (MPLS) label parameters. |

*Note* This keyword is not supported.
The class of service (CoS) number from 0 to 7 to set priority.

**cos**

Name of the pcap file in internal flash (flash:).

**pcap pcap-file-name**

Ensure that the file already exists in flash:.

**Number packet-number**

Specifies the packet number in the pcap file.

**number packet-number**

VLAN id of the dot1q header in the simulated packet. The range is 1 to 4096.

**vlan vlan-id**

---

**Command Modes**

Privileged EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE Gibraltar 16.11.1</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

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

This command supports the following packet types:

- Non-IP packets with any L3 protocol
- ARP packets
- IPv4 packets with any L4 protocol
- IPv4 packets with TCP/UDP/IGMP/ICMP/SCTP payload
- VxLAN packets
- MPLS packets with up to 3 Labels and meta data
- MPLS packets with IPv4/IPv6 payload
- IPv6 packets with TCP/UDP/IGMP/ICMP/SCTP payload

In a stack environment, you can trace packets across the stack irrespective of the number of stack members and topology. The `show platform hardware fed switch switch-number forward interface interface-type interface-number` command consolidates packet-forwarding information of all the stack members on the ingress switch. To achieve this, ensure that the switch number specified in the `switch_num` and `interface-number` arguments are of the input switch and that the number matches.

To trace any particular packet from the captured packets stored in a PCAP file, use the `show platform hardware fed switch forward interface interface-type interface-number pcap pcap-file-name number packet-number data` command.

**Example**

This is an example of output from the `show platform hardware fed switch {switch_num | active | standby} forward interface` command.

```
Device#show platform hardware fed switch active forward interface gigabitEthernet 1/0/35 0000.0022.0055 0000.0055.0066 ipv4 44.44.0.2 55.55.0.2 udp 1222 3333
```
Show forward is running in the background. After completion, syslog will be generated.

*Sep 24 05:57:36.614: %SHFWD-6-PACKET_TRACE_DONE: Switch 1 R0/0: fed: Packet Trace Complete:
Execute (show platform hardware fed switch <> forward last summary|detail)
*Sep 24 05:57:36.614: %SHFWD-6-PACKET_TRACE_FLOW_ID: Switch 1 R0/0: fed: Packet Trace Flow id is 15032385361

<table>
<thead>
<tr>
<th>Related Commands</th>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>monitor capture interface</td>
<td>Configures monitor capture points specifying an attachment point and the packet flow direction.</td>
</tr>
<tr>
<td></td>
<td>monitor capture start</td>
<td>Starts the capture of packet data at a traffic trace point into a buffer.</td>
</tr>
<tr>
<td></td>
<td>monitor capture stop</td>
<td>Stops the capture of packet data at a traffic trace point.</td>
</tr>
<tr>
<td></td>
<td>monitor capture export</td>
<td>Saves the captured packets in the buffer. Use this command to export the monitor capture buffer to a pcap file in flash: that you can use as an input in the show forward with pcap.</td>
</tr>
</tbody>
</table>
**show platform hardware fed switch forward last summary**

To display a summary of packet tracing data from a switch or switches in a stack, use the `show platform hardware fed switch switch_number forward last summary` command.

The output of the `show platform hardware fed switch switch_number forward last summary` command displays all the details about the forwarding decision taken for the packet from the last time the `show forward` command was run.

```
show platform hardware fed switch {switch_number | active | standby} forward last summary
```

### Syntax Description

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>`switch {switch_number</td>
<td>active</td>
</tr>
<tr>
<td></td>
<td><strong>switch_num</strong>—ID of the switch on which the ingress port is present.</td>
</tr>
<tr>
<td></td>
<td><strong>active</strong>—indicates the active switch on which the ingress port is present.</td>
</tr>
<tr>
<td></td>
<td><strong>standby</strong>—indicates the standby switch on which the ingress port is present.</td>
</tr>
</tbody>
</table>

**Note** This keyword is not supported.

<table>
<thead>
<tr>
<th>Command Modes</th>
<th>Privileged EXEC</th>
</tr>
</thead>
</table>

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE Gibraltar 16.11.1</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

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

With Cisco IOS XE Gibraltar 16.10.1, `show platform hardware fed switch forward last summary` command is enhanced to:

- Inject the debug packets from the CPU to simulate the incoming port and packets
- Use the debug packets to trace the packet in hardware data-path to provide forwarding details such as lookup, adjacency, rewrite information, drop decision, outgoing port and so on
- Drop the original packets at egress so as not to transmit the packet to the outgoing port
- Send a copy of all the packets to the CPU and display the details in the packet tracing output

### Example

This is an example of output from the `show platform hardware fed switch {switch_number | active | standby} forward last summary` command.
show platform hardware fed switch active forward last summary

Input Packet Details:

Ethernet dst = 01:00:5e:01:01:02
ts = 00:00:00:03:00:05
type = 0x0

Raw load = '00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

Ingress:
Port : GigabitEthernet1/0/11
Global Port Number : 11
Local Port Number : 11
Asic Port Number : 10
Asic Instance : 1
Vlan : 20
Mapped Vlan ID : 6
STP Instance : 4
BlockForward : 0
BlockLearn : 0
L3 Interface : 39
IPv4 Routing : enabled
IPv6 Routing : enabled
Vrf Id : 0

Adjacency:
Station Index : 3 [SI_DIET_L2]
Destination Index : 18
Rewrite Index : 2
Replication Bit Map : 0x15 ['localData', 'remoteData', 'coreData']

Decision:
Destination Index : 24 [DI_DIET_L2]
Rewrite Index : 2 [RI_L2]
Dest Mod Index : 9 [DMI_IGMP_CTRL_Q]
CPU Map Index : 0 [CMI_NULL]
Forwarding Mode : 0 [Bridging]
Replication Bit Map : ['localData', 'remoteData', 'coreData']
Winner : L2DESTMACVLAN LOOKUP
Qos Label : 65
SGT : 0
DGTID : 0

Egress:
Possible Replication :
Port : GigabitEthernet1/0/11
Port : GigabitEthernet1/0/22
Port : GigabitEthernet2/0/1

Output Port Data :
Port : GigabitEthernet1/0/22
Global Port Number : 22
Local Port Number : 22
Asic Port Number : 21
Asic Instance : 0
Unique RI : 2
Rewrite Type : 1 [L2_BRIDGE]
Mapped Rewrite Type : 1 [L2_BRIDGE]
Vlan : 20
Mapped Vlan ID : 6
Port : GigabitEthernet2/0/1
Global Port Number : 97
Local Port Number : 1
Asic Port Number : 0
Asic Instance : 1
Unique RI : 2
Rewrite Type : 1  [L2_BRIDGE]
Mapped Rewrite Type : 1  [L2_BRIDGE]
Vlan : 20
Mapped Vlan ID : 6

Output Packet Details:
  Port : GigabitEthernet1/0/22
###[ Ethernet ]###
dst = 01:00:5e:01:01:02
src = 00:00:00:03:00:05
type = 0x0
###[ Raw ]###
load = '00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00'
Port : GigabitEthernet2/0/1
###[ Ethernet ]###
dst = 01:00:5e:01:01:02
src = 00:00:00:03:00:05
type = 0x0
###[ Raw ]###
load = '00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00'
*******************************************************************************
show platform resources

To display platform resource information, use the show platform resources command in privileged EXEC mode.

**show platform resources**

This command has no arguments or keywords.

**Command Modes**

- Privileged EXEC (#)

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE Denali 16.1.1</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

The output of this command displays the used memory, which is total memory minus the accurate free memory.

**Example**

The following is sample output from the show platform resources command:

```
Switch# show platform resources
**State Acronym: H - Healthy, W - Warning, C - Critical

<table>
<thead>
<tr>
<th>Resource</th>
<th>State</th>
<th>Usage</th>
<th>Max</th>
<th>Warning</th>
<th>Critical</th>
</tr>
</thead>
<tbody>
<tr>
<td>Control Processor</td>
<td>H</td>
<td>7.20%</td>
<td>100%</td>
<td>90%</td>
<td>95%</td>
</tr>
<tr>
<td>DRAM</td>
<td>H</td>
<td>2701MB(69%)</td>
<td>3883MB</td>
<td>90%</td>
<td>95%</td>
</tr>
</tbody>
</table>
```
show platform software fed switch punt cpuq rates

To display the rate at which packets are punted, including the drops in the punted path, use the `show platform software fed switch punt cpuq rates` command in privileged EXEC mode.

```
show platform software fed switch {switch-number | active | standby} punt cpuq rates
```

**Syntax Description**

- `switch {switch-number | active | standby}`: Displays information about the switch. You have the following options:
  - `switch-number`.
  - `active` — Displays information relating to the active switch.
  - `standby` — Displays information relating to the standby switch, if available.

**Note** This keyword is not supported.

- `punt`: Specifies the punt information.
- `cpuq`: Specifies information about CPU receive queue.
- `rates`: Specifies the rate at which the packets are punted.

**Command Modes**

Privileged EXEC (#)

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE Gibraltar 16.11.1</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

The output of this command displays the rate in packets per second at intervals of 10 seconds, 1 minute and 5 minutes.

**Example**

The following is sample output from the `show platform software fed switch active punt cpuq rates` command.

```
Device#show platform software fed switch active punt cpuq rates

Punt Rate CPU Q Statistics

Packets per second averaged over 10 seconds, 1 min and 5 mins
```

<table>
<thead>
<tr>
<th>Q no</th>
<th>Queue</th>
<th>Rx 10s</th>
<th>Rx 1min</th>
<th>Rx 5min</th>
<th>Drop 10s</th>
<th>Drop 1min</th>
<th>Drop 5min</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>CPU_Q_DOT1X_AUTH</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>CPU_Q_L2_CONTROL</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>CPU_Q_FORUS_TRAFFIC</td>
<td>336</td>
<td>266</td>
<td>320</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
The table below describes the significant fields shown in the display.

**Table 3: show platform software fed switch active punt cpuq rates Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Queue Name</td>
<td>Name of the queue.</td>
</tr>
<tr>
<td>Rx</td>
<td>The rate at which the packets are received per second in 10s, 1 minute and 5 minutes.</td>
</tr>
<tr>
<td>Drop</td>
<td>The rate at which the packets are dropped per second in 10s, 1 minute and 5 minutes.</td>
</tr>
</tbody>
</table>
show platform software fed switch punt packet-capture display

To display packet capture information during high CPU utilization, use the `show platform software fed switch active punt packet-capture display` command in privileged EXEC mode.

```
show platform software fed switch active punt packet-capture display { detailed | hexdump }
```

### Syntax Description

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>`switch{switch-number</td>
<td>active</td>
</tr>
<tr>
<td></td>
<td>• <code>active</code> — Displays information relating to the active switch.</td>
</tr>
<tr>
<td></td>
<td>• <code>standby</code> — Displays information relating to the standby switch, if available.</td>
</tr>
</tbody>
</table>

**Note:** The `standby` keyword is not supported.

<table>
<thead>
<tr>
<th><code>punt</code></th>
<th>Specifies punt information.</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>packet-capture display</code></td>
<td>Specifies information about the captured packet.</td>
</tr>
<tr>
<td><code>detailed</code></td>
<td>Specifies detailed information about the captured packet.</td>
</tr>
<tr>
<td><code>hex-dump</code></td>
<td>Specifies information about the captured packet, in hex format.</td>
</tr>
</tbody>
</table>

### Command Modes

Privileged EXEC (#)

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE Gibraltar 16.11.1</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

The output of this command displays the periodic and persistent logs of CPU-bound packets, inband CPU traffic rates, and running CPU processes when the CPU passes a high CPU utilization threshold.

### Examples

The following is a sample output from the `show platform software fed switch active punt packet-capture display detailed` command:

```
Device# show platform software fed switch active punt packet-capture display detailed
Punt packet capturing: disabled. Buffer wrapping: disabled
Total captured so far: 101 packets. Capture capacity: 4096 packets

interface: GigabitEthernet2/0/2 [if-id: 0x0000000032] (physical)
ether hdr: dest mac: 0100.00cc.00cc, src mac: 2c36.f8fc.4884
ether hdr: ethertype: 0x0032

Doppler Frame Descriptor:
```
Packet Data Dump (length: 68 bytes) :
0100000000000000 C00F402D94510000 0000000000000000 0004004010000000
0000000010000050 000000006D000100 0000000025836200 0000000000000000

interface : GigabitEthernet2/0/2 [if-id: 0x00000032] (physical)
ether hdr : dest mac: 0180.c200.0000, src mac: 2c36.f8fc.4884
ether hdr : ethertype: 0x0026

interface : GigabitEthernet2/0/2 [if-id: 0x00000032] (physical)
ether hdr : dest mac: 0180.c200.0000, src mac: 2c36.f8fc.4884
ether hdr : ethertype: 0x0026

!
show platform software fed switch punt rates interfaces

To display the overall statistics of punt rate for all the interfaces, use the `show platform software fed switch punt rates interfaces` command in privileged EXEC mode.

```
show platform software fed switch {switch-number | active | standby} punt rates interfaces[interface-id]
```

**Syntax Description**

- **switch**: Specifies the switch. You have the following options:
  - `switch-number`.
  - `active`—Displays information relating to the active switch.
  - `standby`—Displays information relating to the standby switch, if available.

- **punt**: Specifies the punt information.

- **rates**: Specifies the rate at which the packets are punted.

- **interfaces[interface-id]**: (Optional) Displays the overall statistics for an interface and also the per-queue configuration for the interface at an interval of 10 seconds.

**Command Modes**

- Privileged EXEC (#)

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE Gibraltar 16.11.1</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

The output displays the punt rates in packets per second at intervals of 10 seconds, 1 minute and 5 minutes.

**Example**

The following is sample output from the `show platform software fed switch active punt rates interfaces` command for all the interfaces.

```
Device#show platform software fed switch active punt rates interfaces

Punt Rate on Interfaces Statistics
Packets per second averaged over 10 seconds, 1 min and 5 mins

<table>
<thead>
<tr>
<th>Interface Name</th>
<th>IF_ID</th>
<th>10s</th>
<th>1min</th>
<th>5min</th>
<th>10s</th>
<th>1min</th>
<th>5min</th>
</tr>
</thead>
</table>
```

---

> Note: This keyword is not supported.
The table below describes the significant fields shown in the display.

**Table 4: show platform software fed switch active punt rates interfaces Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Interface Name</td>
<td>Name of the physical interface.</td>
</tr>
<tr>
<td>IF_ID</td>
<td>ID of the physical interface.</td>
</tr>
<tr>
<td>Rx</td>
<td>The per second rate at which the packets are received in 10s, 1 minute and 5 minutes.</td>
</tr>
<tr>
<td>Drop</td>
<td>The per second rate at which the packets are dropped in 10s, 1 minute and 5 minutes.</td>
</tr>
</tbody>
</table>

The following is sample output from the `show platform software fed switch active punt rates interfaces interface-id` command for a specific interface.

```
Device# show platform software fed switch active punt rates interfaces 0x31
Punt Rate on Single Interfaces Statistics
Interface : Port-channel1 [if_id: 0x31]

Received     | Dropped
--------------|----------
Total 29617   | Total 0
10 sec average: 0 | 10 sec average: 0
1 min average: 0 | 1 min average: 0
5 min average: 0 | 5 min average: 0

Per CPUQ punt stats on the interface (rate averaged over 10s interval)

<table>
<thead>
<tr>
<th>Q</th>
<th>Queue</th>
<th>Recv</th>
<th>Recv</th>
<th>Drop</th>
<th>Drop</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>CPU_Q_DOT1X_AUTH</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>CPU_Q_L2_CONTROL</td>
<td>29519</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>CPU_Q_FORUS_TRAFFIC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>CPU_Q_ICMP_GEN</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>CPU_Q_ROUTING_CONTROL</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>CPU_Q_FORUS_ADDR_RESOLUTION</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>CPU_Q_ICMP_REDIRECT</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>7</td>
<td>CPU_Q_INTER_FED_TRAFFIC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>8</td>
<td>CPU_Q_L2LVX_CONTROL_PKT</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>9</td>
<td>CPU_Q_EWLC_CONTROL</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>10</td>
<td>CPU_Q_EWLC_DATA</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>11</td>
<td>CPU_Q_L2LVX_DATA_PKT</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>12</td>
<td>CPU_Q_BROADCAST</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>13</td>
<td>CPU_Q_LEARNING_CACHE_OVFL</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>14</td>
<td>CPU_Q_SN_FORWARDING</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>15</td>
<td>CPU_Q_TOPOLOGY_CONTROL</td>
<td>98</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>16</td>
<td>CPU_Q_PROTO_SNOOPING</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>17</td>
<td>CPU_Q_DHCP_SNOOPING</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>18</td>
<td>CPU_Q_TRANSIT_TRAFFIC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>19</td>
<td>CPU_Q_RPF_FAILED</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
```
The table below describes the significant fields shown in the display.

**Table 5: show platform software fed switch punt rates interfaces interface-id Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Queue Name</td>
<td>Name of the queue.</td>
</tr>
<tr>
<td>Recv Total</td>
<td>Total number of packets received.</td>
</tr>
<tr>
<td>Recv Rate</td>
<td>Per second rate at which the packets are received.</td>
</tr>
<tr>
<td>Drop Total</td>
<td>Total number of packets dropped.</td>
</tr>
<tr>
<td>Drop Rate</td>
<td>Per second rate at which the packets are dropped.</td>
</tr>
</tbody>
</table>
show platform software ilpower

To display the inline power details of all the PoE ports on the device, use the `show platform software ilpower` command in privileged EXEC mode.

```plaintext
show platform software ilpower {details | port (GigabitEthernet interface-number } | system slot-number }
```

**Syntax Description**

- **details**
  - Displays inline power details for all the interfaces.
- **port**
  - Displays inline power port configuration.
- **GigabitEthernet interface-number**
  - The GigabitEthernet interface number. Values range from 0 to 9.
- **system slot-number**
  - Displays inline power system configuration.

**Command Modes**

Privileged EXEC (#)

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE Denali 16.3.2</td>
<td>This command was modified. The keyword <code>details</code> argument was added.</td>
</tr>
<tr>
<td>Cisco IOS XE Denali 16.1.1</td>
<td>The command was introduced.</td>
</tr>
</tbody>
</table>

**Examples**

The following is sample output from the `show platform software ilpower details` command:

```
Device# show platform software ilpower details
ILP Port Configuration for interface Gi1/0/1
  Initialization Done: Yes
  ILP Supported: Yes
  ILP Enabled: Yes
  POST: Yes
  Detect On: No
  Powered Device Detected No
  Powered Device Class Done No
  Cisco Powered Device: No
  Power is On: No
  Power Denied: No
  Powered Device Type: Null
  Powered Device Class: Null
  Power State: NULL
  Current State: NGWC_ILP_DETECTING_S
  Previous State: NGWC_ILP_SHUT_OFF_S
  Requested Power in milli watts: 0
  Short Circuit Detected: 0
  Short Circuit Count: 0
  Cisco Powered Device Detect Count: 0
  Spare Pair mode: 0
    IEEE Detect: Stopped
    IEEE Short: Stopped
    Link Down: Stopped
    Voltage sense: Stopped
  Spare Pair Architecture: 1
```
show platform software ilpower

Signal Pair Power allocation in milli watts: 0
Spare Pair Power On: 0
Powered Device power state: 0
Timer:
  Power Good: Stopped
  Power Denied: Stopped
  Cisco Powered Device Detect: Stopped
show platform software memory

To display memory information for a specified switch, use the `show platform software memory` command in privileged EXEC mode.

```
show platform software memory  [{chunk | database | messaging}] process slot
```

<table>
<thead>
<tr>
<th>Syntax Description</th>
<th>chunk</th>
<th>database</th>
<th>messaging</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>(Optional) Displays chunk memory information for the specified process.</td>
<td>(Optional) Displays database memory information for the specified process.</td>
<td>(Optional) Displays messaging memory information for the specified process. The information displayed is for internal debugging purposes only.</td>
</tr>
</tbody>
</table>


show platform software memory

process
Level that is being set. Options include:

- **bt-logger**—The Binary-Tracing Logger process.
- **btrace-manager**—The Btrace Manager process.
- **chassis-manager**—The Chassis Manager process.
- **cli-agent**—The CLI Agent process.
- **cmm**—The CMM process.
- **dbm**—The Database Manager process.
- **dmiauthd**—The DMI Authentication Daemon process.
- **emd**—The Environmental Monitoring process.
- **fed**—The Forwarding Engine Driver process.
- **forwarding-manager**—The Forwarding Manager process.
- **geo**—The Geo Manager process.
- **gnmi**—The GNMI process.
- **host-manager**—The Host Manager process.
- **interface-manager**—The Interface Manager process.
- **iomd**—The Input/Output Module daemon (IOMd) process.
- **ios**—The IOS process.
- **iox-manager**—The IOX Manager process.
- **license-manager**—The License Manager process.
- **logger**—The Logging Manager process.
- **mdt-pubd**—The Model Defined Telemetry Publisher process.
- **ndbman**—The Netconf DataBase Manager process.
- **nesd**—The Network Element Synchronizer Daemon process.
- **nginx**—The Nginx Webserver process.
- **nif_mgr**—The NIF Manager process.
- **platform-mgr**—The Platform Manager process.
- **pluggable-services**—The Pluggable Services process.
- **replication-mgr**—The Replication Manager process.
- **shell-manager**—The Shell Manager process.
- **sif**—The Stack Interface (SIF) Manager process.
- **smd**—The Session Manager process.
- **stack-mgr**—The Stack Manager process.
• **syncfd**—The SyncmDaemon process.

• **table-manager**—The Table Manager Server.

• **thread-test**—The Multithread Manager process.

• **virt-manager**—The Virtualization Manager process.

---

**slot**

Hardware slot where the process for which the level is set, is running. Options include:

• **number**—Number of the SIP slot of the hardware module where the level is set. For instance, if you want to specify the SIP in SIP slot 2 of the switch, enter 2.

• **SIP-slot / SPA-bay**—Number of the SIP switch slot and the number of the shared port adapter (SPA) bay of that SIP. For instance, if you want to specify the SPA in bay 2 of the SIP in switch slot 3, enter 3/2.

• **F0**—The Embedded Service Processor slot 0.

• **FP active**—The active Embedded Service Processor.

• **R0**—The route processor in slot 0.

• **RP active**—The active route processor.

• **switch <number>**—The switch, with its number specified.

• **switch active**—The active switch.

• **switch standby**—The standby switch.

  • **number**—Number of the SIP slot of the hardware module where the level is set. For instance, if you want to specify the SIP in SIP slot 2 of the switch, enter 2.

  • **SIP-slot / SPA-bay**—Number of the SIP switch slot and the number of the shared port adapter (SPA) bay of that SIP. For instance, if you want to specify the SPA in bay 2 of the SIP in switch slot 3, enter 3/2.

• **F0**—The Embedded Service Processor in slot 0.

• **FP active**—The active Embedded Service Processor.

• **R0**—The route processor in slot 0.

• **RP active**—The active route processor.

---

**Command Default**

No default behavior or values.

**Command Modes**

Privileged EXEC (#)
The following is a sample output displaying the abbreviated (brief keyword) memory information for the Forwarding Manager process for Cisco Catalyst 3000 Series ESP slot 0:

```
Device# show platform software memory forwarding-manager switch 1 fp active brief

<table>
<thead>
<tr>
<th>module</th>
<th>allocated</th>
<th>requested</th>
<th>allocs</th>
<th>frees</th>
</tr>
</thead>
<tbody>
<tr>
<td>Summary</td>
<td>5702540</td>
<td>5619788</td>
<td>121888</td>
<td>116716</td>
</tr>
<tr>
<td>AOM object</td>
<td>1920374</td>
<td>1920310</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>AOM links array</td>
<td>880379</td>
<td>880315</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>smc_message</td>
<td>819575</td>
<td>819511</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>AOM update state</td>
<td>640380</td>
<td>640316</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>dpidb-config</td>
<td>208776</td>
<td>203544</td>
<td>351</td>
<td>24</td>
</tr>
<tr>
<td>fman-infra-avl</td>
<td>178016</td>
<td>153680</td>
<td>1521</td>
<td>0</td>
</tr>
<tr>
<td>AOM batch</td>
<td>152373</td>
<td>152309</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>AOM asynchronous conte</td>
<td>128388</td>
<td>128324</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>AOM basic data</td>
<td>124824</td>
<td>124760</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>eventutil</td>
<td>118939</td>
<td>118299</td>
<td>50</td>
<td>10</td>
</tr>
<tr>
<td>AOM tree node</td>
<td>96465</td>
<td>96385</td>
<td>5</td>
<td>0</td>
</tr>
<tr>
<td>AOM tree root</td>
<td>72377</td>
<td>72313</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>acl</td>
<td>36090</td>
<td>31914</td>
<td>504</td>
<td>0</td>
</tr>
<tr>
<td>fman-infra-ipc</td>
<td>35326</td>
<td>24366</td>
<td>115097</td>
<td>114412</td>
</tr>
<tr>
<td>AOM uplink update node</td>
<td>32386</td>
<td>32322</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>unknown</td>
<td>30528</td>
<td>23808</td>
<td>424</td>
<td>4</td>
</tr>
<tr>
<td>uipeer</td>
<td>27232</td>
<td>27152</td>
<td>5</td>
<td>0</td>
</tr>
<tr>
<td>fman-infra-qos</td>
<td>26872</td>
<td>24712</td>
<td>164</td>
<td>29</td>
</tr>
<tr>
<td>cce-class</td>
<td>19427</td>
<td>15411</td>
<td>251</td>
<td>0</td>
</tr>
<tr>
<td>l2 control protocol</td>
<td>15472</td>
<td>12896</td>
<td>325</td>
<td>164</td>
</tr>
<tr>
<td>fman-infra-cce</td>
<td>15272</td>
<td>13576</td>
<td>106</td>
<td>0</td>
</tr>
<tr>
<td>smc_channel</td>
<td>15223</td>
<td>15159</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>unknown</td>
<td>14208</td>
<td>8736</td>
<td>447</td>
<td>105</td>
</tr>
<tr>
<td>chunk</td>
<td>12513</td>
<td>12033</td>
<td>33</td>
<td>3</td>
</tr>
<tr>
<td>cce-bind</td>
<td>8496</td>
<td>7552</td>
<td>82</td>
<td>23</td>
</tr>
<tr>
<td>MATM mac entry</td>
<td>8040</td>
<td>5928</td>
<td>544</td>
<td>412</td>
</tr>
<tr>
<td>adj</td>
<td>7064</td>
<td>6312</td>
<td>157</td>
<td>110</td>
</tr>
<tr>
<td>route-pfx</td>
<td>6116</td>
<td>5412</td>
<td>157</td>
<td>113</td>
</tr>
<tr>
<td>filter_rules</td>
<td>4912</td>
<td>4896</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>fman-infra-dpidb</td>
<td>4130</td>
<td>2338</td>
<td>112</td>
<td>0</td>
</tr>
<tr>
<td>SMC Buffer</td>
<td>3794</td>
<td>3202</td>
<td>43</td>
<td>6</td>
</tr>
<tr>
<td>urpf-list</td>
<td>3028</td>
<td>2100</td>
<td>85</td>
<td>27</td>
</tr>
<tr>
<td>lookup</td>
<td>2480</td>
<td>2160</td>
<td>30</td>
<td>10</td>
</tr>
<tr>
<td>MATM mac table</td>
<td>2432</td>
<td>1600</td>
<td>148</td>
<td>96</td>
</tr>
<tr>
<td>cdllib</td>
<td>1688</td>
<td>1672</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>route-tbl</td>
<td>1600</td>
<td>1264</td>
<td>21</td>
<td>0</td>
</tr>
<tr>
<td>FNF Flowdef</td>
<td>1492</td>
<td>1460</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>acl-ref</td>
<td>1120</td>
<td>1024</td>
<td>8</td>
<td>2</td>
</tr>
<tr>
<td>cgm-lib</td>
<td>1120</td>
<td>880</td>
<td>410</td>
<td>395</td>
</tr>
<tr>
<td>pbr_if_cfg</td>
<td>1088</td>
<td>976</td>
<td>205</td>
<td>198</td>
</tr>
<tr>
<td>FNF Monitor</td>
<td>1048</td>
<td>1032</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>pbr_routemap</td>
<td>960</td>
<td>864</td>
<td>18</td>
<td>12</td>
</tr>
</tbody>
</table>
```

The following table describes the significant fields shown in the display.
### Table 6: show platform software memory brief Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>module</td>
<td>Name of submodule.</td>
</tr>
<tr>
<td>allocated</td>
<td>Memory, allocated in bytes.</td>
</tr>
<tr>
<td>requested</td>
<td>Number of bytes requested by application.</td>
</tr>
<tr>
<td>allocs</td>
<td>Number of discrete allocation event attempts.</td>
</tr>
<tr>
<td>frees</td>
<td>Number of free events.</td>
</tr>
</tbody>
</table>
show platform software process list

To display the list of running processes on a platform, use the `show platform software process list` command in privileged EXEC mode.

```
show platform software process list switch {switch-number | active | standby} {0 | F0 | R0}
[[name process-name | process-id process-ID | sort memory | summary]}
```

### Syntax Description

| switch switch-number | Displays information about the switch. Valid values for `switch-number` argument are from 0 to 9. |
| active | Displays information about the active instance of the switch. |
| standby | Displays information about the standby instance of the switch. |
| 0 | Displays information about the shared port adapters (SPA) Interface Processor slot 0. |
| F0 | Displays information about the Embedded Service Processor (ESP) slot 0. |
| R0 | Displays information about the Route Processor (RP) slot 0. |
| name process-name | (Optional) Displays information about the specified process. Enter the process name. |
| process-id process-ID | (Optional) Displays information about the specified process ID. Enter the process ID. |
| sort | (Optional) Displays information sorted according to processes. |
| memory | (Optional) Displays information sorted according to memory. |
| summary | (Optional) Displays a summary of the process memory of the host device. |

### Command Modes

Privileged EXE (#)

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE Denali 16.1.1</td>
<td>The command was introduced.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

Prior to Cisco IOS XE Denali 16.3.1, the Free Memory displayed in the command output was obtained from the underlying Linux kernel. This value was not accurate because some memory chunks that was available for use was not considered as free memory.

In Cisco IOS XE Denali 16.3.1, the free memory is accurately computed and displayed in the Free Memory field of the command output.

### Examples

The following is sample output from the `show platform software process list switch active R0` command:

```
Switch# show platform software process list switch active R0 summary
Total number of processes: 278
```
show platform software process list

Running : 2
Sleeping : 276
Disk sleeping : 0
Zombies : 0
Stopped : 0
Paging : 0

Up time : 8318
Idle time : 0
User time : 216809
Kernel time : 78931

Virtual memory : 12933324800
Pages resident : 634061
Major page faults: 2228
Minor page faults: 3491744

Architecture : mips64

Memory (kB)
  Physical : 3976852
  Total : 3976852
  Used : 2766952
  Free : 12098900
  Active : 2141344
  Inactive : 1589672
  Inact-dirty : 0
  Inact-clean : 0
  Dirty : 4
  AnonPages : 1306800
  Bounce : 0
  Cached : 1984688
  Commit Limit : 1988424
  Committed As : 3358528
  High Total : 0
  High Free : 0
  Low Total : 3976852
  Low Free : 1209900
  Mapped : 520528
  NFS Unstable : 0
  Page Tables : 17328
  Slab : 0
  VMmalloc Chunk : 1069542588
  VMmalloc Total : 1069547512
  VMmalloc Used : 2588
  Writeback : 0
  HugePages Total: 0
  HugePages Free : 0
  HugePages Rsvd : 0
  HugePage Size : 2048

Swap (kB)
  Total : 0
  Used : 0
  Free : 0
  Cached : 0

Buffers (kB) : 439528

Load Average
  1-Min : 1.13
  5-Min : 1.18
  15-Min : 0.92
The following is sample output from the `show platform software process list switch active R0` command:

```
Device# show platform software process list switch active R0

<table>
<thead>
<tr>
<th>Name</th>
<th>Pid</th>
<th>PPid</th>
<th>Group Id</th>
<th>Status</th>
<th>Priority</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>systemd</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>S</td>
<td>20</td>
<td>7892</td>
</tr>
<tr>
<td>kthreadd</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>S</td>
<td>20</td>
<td>0</td>
</tr>
<tr>
<td>ksoftirqd/0</td>
<td>3</td>
<td>2</td>
<td>0</td>
<td>S</td>
<td>20</td>
<td>0</td>
</tr>
<tr>
<td>kworker/0:0H</td>
<td>5</td>
<td>2</td>
<td>0</td>
<td>S</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>rcu_sched</td>
<td>7</td>
<td>2</td>
<td>0</td>
<td>S</td>
<td>20</td>
<td>0</td>
</tr>
<tr>
<td>rcu_bh</td>
<td>8</td>
<td>2</td>
<td>0</td>
<td>S</td>
<td>20</td>
<td>0</td>
</tr>
<tr>
<td>migration/0</td>
<td>9</td>
<td>2</td>
<td>0</td>
<td>S</td>
<td>4294967196</td>
<td>0</td>
</tr>
<tr>
<td>migration/1</td>
<td>10</td>
<td>2</td>
<td>0</td>
<td>S</td>
<td>4294967196</td>
<td>0</td>
</tr>
<tr>
<td>ksoftirqd/1</td>
<td>11</td>
<td>2</td>
<td>0</td>
<td>S</td>
<td>20</td>
<td>0</td>
</tr>
<tr>
<td>kworker/1:0H</td>
<td>13</td>
<td>2</td>
<td>0</td>
<td>S</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>migration/2</td>
<td>14</td>
<td>2</td>
<td>0</td>
<td>S</td>
<td>4294967196</td>
<td>0</td>
</tr>
<tr>
<td>ksoftirqd/2</td>
<td>15</td>
<td>2</td>
<td>0</td>
<td>S</td>
<td>20</td>
<td>0</td>
</tr>
<tr>
<td>kworker/2:0H</td>
<td>17</td>
<td>2</td>
<td>0</td>
<td>S</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>systemd-journal</td>
<td>221</td>
<td>1</td>
<td>221</td>
<td>S</td>
<td>20</td>
<td>4460</td>
</tr>
<tr>
<td>kworker/1:3</td>
<td>246</td>
<td>2</td>
<td>0</td>
<td>S</td>
<td>20</td>
<td>0</td>
</tr>
<tr>
<td>systemd-udevd</td>
<td>253</td>
<td>1</td>
<td>253</td>
<td>S</td>
<td>20</td>
<td>5648</td>
</tr>
<tr>
<td>kvm-irqfd-clean</td>
<td>617</td>
<td>2</td>
<td>0</td>
<td>S</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>scsi_eh_6</td>
<td>620</td>
<td>2</td>
<td>0</td>
<td>S</td>
<td>20</td>
<td>0</td>
</tr>
<tr>
<td>acsi_tmf_6</td>
<td>621</td>
<td>2</td>
<td>0</td>
<td>S</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>usb-storage</td>
<td>622</td>
<td>2</td>
<td>0</td>
<td>S</td>
<td>20</td>
<td>0</td>
</tr>
<tr>
<td>acsi_eh_7</td>
<td>625</td>
<td>2</td>
<td>0</td>
<td>S</td>
<td>20</td>
<td>0</td>
</tr>
<tr>
<td>acsi_tmf_7</td>
<td>626</td>
<td>2</td>
<td>0</td>
<td>S</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>usb-storage</td>
<td>627</td>
<td>2</td>
<td>0</td>
<td>S</td>
<td>20</td>
<td>0</td>
</tr>
<tr>
<td>kworker/7:1</td>
<td>630</td>
<td>2</td>
<td>0</td>
<td>S</td>
<td>20</td>
<td>0</td>
</tr>
<tr>
<td>bioset</td>
<td>631</td>
<td>2</td>
<td>0</td>
<td>S</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>kworker/3:1H</td>
<td>648</td>
<td>2</td>
<td>0</td>
<td>S</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>kworker/0:1H</td>
<td>667</td>
<td>2</td>
<td>0</td>
<td>S</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>kworker/1:1H</td>
<td>668</td>
<td>2</td>
<td>0</td>
<td>S</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>bioset</td>
<td>669</td>
<td>2</td>
<td>0</td>
<td>S</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>kworker/6:2</td>
<td>698</td>
<td>2</td>
<td>0</td>
<td>S</td>
<td>20</td>
<td>0</td>
</tr>
<tr>
<td>kworker/2:2</td>
<td>699</td>
<td>2</td>
<td>0</td>
<td>S</td>
<td>20</td>
<td>0</td>
</tr>
<tr>
<td>kworker/2:1H</td>
<td>703</td>
<td>2</td>
<td>0</td>
<td>S</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>kworker/7:1H</td>
<td>748</td>
<td>2</td>
<td>0</td>
<td>S</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>kworker/5:1H</td>
<td>749</td>
<td>2</td>
<td>0</td>
<td>S</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>kworker/6:1H</td>
<td>754</td>
<td>2</td>
<td>0</td>
<td>S</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>kworker/7:2</td>
<td>779</td>
<td>2</td>
<td>0</td>
<td>S</td>
<td>20</td>
<td>0</td>
</tr>
<tr>
<td>auditd</td>
<td>838</td>
<td>1</td>
<td>838</td>
<td>S</td>
<td>16</td>
<td>2564</td>
</tr>
</tbody>
</table>
```

The table below describes the significant fields shown in the displays.

### Table 7: `show platform software process list` Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Displays the command name associated with the process. Different threads in the same process may have different command values.</td>
</tr>
<tr>
<td>Pid</td>
<td>Displays the process ID that is used by the operating system to identify and keep track of the processes.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>PPid</td>
<td>Displays process ID of the parent process.</td>
</tr>
<tr>
<td>Group Id</td>
<td>Displays the group ID</td>
</tr>
<tr>
<td>Status</td>
<td>Displays the process status in human readable form.</td>
</tr>
<tr>
<td>Priority</td>
<td>Displays the negated scheduling priority.</td>
</tr>
<tr>
<td>Size</td>
<td>Prior to Cisco IOS XE Gibraltar 16.10.1:</td>
</tr>
<tr>
<td></td>
<td>Displays Virtual Memory size.</td>
</tr>
<tr>
<td></td>
<td>From Cisco IOS XE Gibraltar 16.10.1 onwards:</td>
</tr>
<tr>
<td></td>
<td>Displays the Resident Set Size (RSS) that shows how much memory is allocated to that process in the RAM.</td>
</tr>
</tbody>
</table>
show platform software process memory

To display the amount of memory used by each system process, use the **show platform software process memory** command in privileged EXEC mode.

```
show platform process memory
switch { switch-number | active | standby | 0 | F0 | FP | R0 } { all | sorted | virtual | sorted } [ name process-name { maps | smaps [ summary ] } | process-id process-id { maps | smaps { summary } } ]
```

**Syntax Description**

<table>
<thead>
<tr>
<th>switch switch-number</th>
<th>Displays information about the switch. Enter the switch number.</th>
</tr>
</thead>
<tbody>
<tr>
<td>active</td>
<td>Specifies the active instance of the device.</td>
</tr>
<tr>
<td>standby</td>
<td>Specifies the standby instance of the device.</td>
</tr>
<tr>
<td>0</td>
<td>Specifies the Shared Port Adapter (SPA) Interface Processor slot 0.</td>
</tr>
<tr>
<td>F0</td>
<td>Specifies the Embedded Service Processor (ESP) slot 0.</td>
</tr>
<tr>
<td>FP</td>
<td>Specifies the Embedded Service Processor (ESP).</td>
</tr>
<tr>
<td>R0</td>
<td>Specifies the Route Processor (RP) slot 0.</td>
</tr>
<tr>
<td>all</td>
<td>Lists all processes.</td>
</tr>
<tr>
<td>sorted</td>
<td>(Optional) Sorts the output based on Resident Set Size (RSS).</td>
</tr>
<tr>
<td>virtual</td>
<td>(Optional) Specifies virtual memory.</td>
</tr>
<tr>
<td>name process-name</td>
<td>Specifies a process name.</td>
</tr>
<tr>
<td>maps</td>
<td>Specifies the memory maps of a process.</td>
</tr>
<tr>
<td>smaps summary</td>
<td>Specifies the smaps summary of a process.</td>
</tr>
<tr>
<td>process-id process-id</td>
<td>Specifies a process identifier.</td>
</tr>
</tbody>
</table>

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE Gibraltar 16.11.1</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Command Modes**

Privileged EXEC(#) 

**Examples:**

The following is a sample output from the **show platform software process memory active R0 all** command:
Device# show platform software process memory switch active R0 all

<table>
<thead>
<tr>
<th>Pid</th>
<th>RSS</th>
<th>PSS</th>
<th>Heap</th>
<th>Shared</th>
<th>Private</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>4876</td>
<td>3229</td>
<td>1064</td>
<td>1808</td>
<td>3068</td>
<td>systemd</td>
</tr>
<tr>
<td>118</td>
<td>3184</td>
<td>1327</td>
<td>132</td>
<td>2352</td>
<td>832</td>
<td>systemd-journal</td>
</tr>
<tr>
<td>159</td>
<td>3008</td>
<td>1191</td>
<td>396</td>
<td>1996</td>
<td>1012</td>
<td>systemd-udevd</td>
</tr>
<tr>
<td>407</td>
<td>3192</td>
<td>1262</td>
<td>132</td>
<td>2196</td>
<td>996</td>
<td>dbus-daemon</td>
</tr>
<tr>
<td>3406</td>
<td>4772</td>
<td>3064</td>
<td>264</td>
<td>1940</td>
<td>2832</td>
<td>virtlogd</td>
</tr>
<tr>
<td>3411</td>
<td>5712</td>
<td>3474</td>
<td>2964</td>
<td>2344</td>
<td>3368</td>
<td>droputil.sh</td>
</tr>
<tr>
<td>3416</td>
<td>2588</td>
<td>358</td>
<td>132</td>
<td>2336</td>
<td>252</td>
<td>libvirtd.sh</td>
</tr>
<tr>
<td>3420</td>
<td>5708</td>
<td>3484</td>
<td>2976</td>
<td>2308</td>
<td>3000</td>
<td>reflector.sh</td>
</tr>
<tr>
<td>3424</td>
<td>1804</td>
<td>263</td>
<td>132</td>
<td>1632</td>
<td>172</td>
<td>xinetd</td>
</tr>
<tr>
<td>3425</td>
<td>964</td>
<td>118</td>
<td>132</td>
<td>872</td>
<td>92</td>
<td>sleep</td>
</tr>
<tr>
<td>3434</td>
<td>3060</td>
<td>844</td>
<td>528</td>
<td>2304</td>
<td>756</td>
<td>com.sh</td>
</tr>
<tr>
<td>3442</td>
<td>2068</td>
<td>606</td>
<td>132</td>
<td>1604</td>
<td>464</td>
<td>rpcbind</td>
</tr>
<tr>
<td>3485</td>
<td>2380</td>
<td>845</td>
<td>132</td>
<td>1636</td>
<td>744</td>
<td>rpc.statd</td>
</tr>
<tr>
<td>3486</td>
<td>1632</td>
<td>338</td>
<td>132</td>
<td>1348</td>
<td>284</td>
<td>boohelper_evt.</td>
</tr>
<tr>
<td>3493</td>
<td>1136</td>
<td>156</td>
<td>132</td>
<td>1004</td>
<td>132</td>
<td>inotifywait</td>
</tr>
<tr>
<td>3504</td>
<td>2048</td>
<td>753</td>
<td>132</td>
<td>1372</td>
<td>676</td>
<td>rpc.mountd</td>
</tr>
<tr>
<td>3584</td>
<td>2868</td>
<td>620</td>
<td>36</td>
<td>2384</td>
<td>484</td>
<td>rotee</td>
</tr>
<tr>
<td>3649</td>
<td>1032</td>
<td>116</td>
<td>132</td>
<td>944</td>
<td>88</td>
<td>sleep</td>
</tr>
<tr>
<td>3705</td>
<td>2784</td>
<td>613</td>
<td>36</td>
<td>2296</td>
<td>488</td>
<td>rotee</td>
</tr>
<tr>
<td>3718</td>
<td>2856</td>
<td>610</td>
<td>36</td>
<td>2376</td>
<td>480</td>
<td>rotee</td>
</tr>
<tr>
<td>3759</td>
<td>1292</td>
<td>184</td>
<td>132</td>
<td>1136</td>
<td>156</td>
<td>inotifywait</td>
</tr>
<tr>
<td>3787</td>
<td>4256</td>
<td>2040</td>
<td>1640</td>
<td>2300</td>
<td>1956</td>
<td>iptbl.sh</td>
</tr>
<tr>
<td>3894</td>
<td>2948</td>
<td>637</td>
<td>36</td>
<td>2460</td>
<td>488</td>
<td>rotee</td>
</tr>
<tr>
<td>4017</td>
<td>1380</td>
<td>176</td>
<td>132</td>
<td>1236</td>
<td>144</td>
<td>inotifywait</td>
</tr>
<tr>
<td>4866</td>
<td>1820</td>
<td>287</td>
<td>132</td>
<td>1624</td>
<td>196</td>
<td>xinetd</td>
</tr>
<tr>
<td>5887</td>
<td>1692</td>
<td>257</td>
<td>132</td>
<td>1508</td>
<td>184</td>
<td>xinetd</td>
</tr>
<tr>
<td>5891</td>
<td>7248</td>
<td>4984</td>
<td>4584</td>
<td>2348</td>
<td>4900</td>
<td>rollback_timer.</td>
</tr>
<tr>
<td>5893</td>
<td>1764</td>
<td>257</td>
<td>132</td>
<td>1588</td>
<td>176</td>
<td>xinetd</td>
</tr>
<tr>
<td>6031</td>
<td>2804</td>
<td>601</td>
<td>36</td>
<td>2332</td>
<td>472</td>
<td>rotee</td>
</tr>
<tr>
<td>6037</td>
<td>1228</td>
<td>163</td>
<td>132</td>
<td>1092</td>
<td>136</td>
<td>inotifywait</td>
</tr>
<tr>
<td>6077</td>
<td>4736</td>
<td>3389</td>
<td>2992</td>
<td>1368</td>
<td>3368</td>
<td>psvp.sh</td>
</tr>
<tr>
<td>6115</td>
<td>1620</td>
<td>476</td>
<td>36</td>
<td>1152</td>
<td>468</td>
<td>rotee</td>
</tr>
<tr>
<td>6122</td>
<td>624</td>
<td>149</td>
<td>132</td>
<td>480</td>
<td>144</td>
<td>inotifywait</td>
</tr>
<tr>
<td>6127</td>
<td>5440</td>
<td>4077</td>
<td>3680</td>
<td>1384</td>
<td>4056</td>
<td>psvp.sh</td>
</tr>
<tr>
<td>6165</td>
<td>1736</td>
<td>592</td>
<td>36</td>
<td>1152</td>
<td>584</td>
<td>rotee</td>
</tr>
<tr>
<td>6245</td>
<td>624</td>
<td>149</td>
<td>132</td>
<td>480</td>
<td>144</td>
<td>inotifywait</td>
</tr>
<tr>
<td>6353</td>
<td>2592</td>
<td>1260</td>
<td>924</td>
<td>1352</td>
<td>1240</td>
<td>pman.sh</td>
</tr>
<tr>
<td>6470</td>
<td>1632</td>
<td>488</td>
<td>36</td>
<td>1152</td>
<td>480</td>
<td>rotee</td>
</tr>
<tr>
<td>6499</td>
<td>2588</td>
<td>1262</td>
<td>924</td>
<td>1348</td>
<td>1240</td>
<td>pman.sh</td>
</tr>
<tr>
<td>6666</td>
<td>1640</td>
<td>496</td>
<td>36</td>
<td>1152</td>
<td>488</td>
<td>rotee</td>
</tr>
<tr>
<td>6718</td>
<td>2584</td>
<td>1258</td>
<td>800</td>
<td>1348</td>
<td>1236</td>
<td>pman.sh</td>
</tr>
<tr>
<td>6736</td>
<td>8360</td>
<td>7020</td>
<td>6640</td>
<td>1360</td>
<td>7000</td>
<td>auto_upgrade_cl</td>
</tr>
<tr>
<td>6909</td>
<td>1636</td>
<td>492</td>
<td>36</td>
<td>1152</td>
<td>484</td>
<td>rotee</td>
</tr>
<tr>
<td>6955</td>
<td>2588</td>
<td>1262</td>
<td>928</td>
<td>1348</td>
<td>1240</td>
<td>pman.sh</td>
</tr>
<tr>
<td>7029</td>
<td>2196</td>
<td>679</td>
<td>40</td>
<td>1552</td>
<td>644</td>
<td>auto_upgrade_se</td>
</tr>
<tr>
<td>7149</td>
<td>1636</td>
<td>492</td>
<td>36</td>
<td>1152</td>
<td>484</td>
<td>rotee</td>
</tr>
<tr>
<td>7224</td>
<td>13200</td>
<td>4595</td>
<td>48</td>
<td>9368</td>
<td>3832</td>
<td>bt_logger</td>
</tr>
<tr>
<td>7295</td>
<td>2588</td>
<td>1262</td>
<td>800</td>
<td>1348</td>
<td>1240</td>
<td>pman.sh</td>
</tr>
</tbody>
</table>

The table below describes the significant fields shown in the displays.
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PID</td>
<td>Displays the process ID that is used by the operating system to identify and keep track of the processes.</td>
</tr>
<tr>
<td>RSS</td>
<td>Displays the Resident Set Size (in kilobytes (KB)) that shows how much memory is allocated to that process in the RAM.</td>
</tr>
<tr>
<td>PSS</td>
<td>Displays the Proportional Set Size of a process. This is the count of pages it has in memory, where each page is divided by the number of processes sharing it.</td>
</tr>
<tr>
<td>Heap</td>
<td>Displays where all user-allocated memory is located.</td>
</tr>
<tr>
<td>Shared</td>
<td>Shared clean + Shared dirty</td>
</tr>
<tr>
<td>Private</td>
<td>Private clean + Private dirty</td>
</tr>
<tr>
<td>Name</td>
<td>Displays the command name associated with the process. Different threads in the same process may have different command values.</td>
</tr>
</tbody>
</table>
show platform software process slot switch

To display platform software process switch information, use the `show platform software process slot switch` command in privileged EXEC mode.

```
show platform software process slot switch {switch-number | active | standby} {0 | F0 | R0} monitor [{cycles no-of-times [{interval delay [{lines number}]}]}]
```

**Syntax Description**

- **switch-number**: Switch number.
- **active**: Specifies the active instance.
- **standby**: Specifies the standby instance.
- **0**: Specifies the shared port adapter (SPA) interface processor slot 0.
- **F0**: Specifies the Embedded Service Processor (ESP) slot 0.
- **R0**: Specifies the Route Processor (RP) slot 0.
- **monitor**: Monitors the running processes.
- **cycles no-of-times**: (Optional) Sets the number of times to run monitor command. Valid values are from 1 to 4294967295. The default is 5.
- **interval delay**: (Optional) Sets a delay after each. Valid values are from 0 to 300. The default is 3.
- **lines number**: (Optional) Sets the number of lines of output displayed. Valid values are from 0 to 512. The default is 0.

**Command Modes**

Privileged EXEC (#)

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE Denali 16.1.1</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

The output of the `show platform software process slot switch` and `show processes cpu platform monitor location` commands display the output of the Linux `top` command. The output of these commands display Free memory and Used memory as displayed by the Linux `top` command. The values displayed for the Free memory and Used memory by these commands do not match the values displayed by the output of other platform-memory related CLIs.

**Examples**

The following is sample output from the `show platform software process slot switch active R0 monitor` command:
Switch# show platform software process slot switch active R0 monitor

top - 00:01:52 up 1 day, 11:20, 0 users, load average: 0.50, 0.68, 0.83
Tasks: 311 total, 2 running, 309 sleeping, 0 stopped, 0 zombie
Cpu(s): 7.4%us, 3.3%sy, 0.0%ni, 89.2%id, 0.0%wa, 0.0%hi, 0.1%si, 0.0%st
Mem: 39768k total, 3955036k used, 21808k free, 419312k buffers
Swap: 0k total, 0k used, 0k free, 1946764k cached

<table>
<thead>
<tr>
<th>PID</th>
<th>USER</th>
<th>PR</th>
<th>NI</th>
<th>VIRT</th>
<th>RES</th>
<th>SHR</th>
<th>S</th>
<th>%CPU</th>
<th>%MEM</th>
<th>TIME+</th>
<th>COMMAND</th>
</tr>
</thead>
<tbody>
<tr>
<td>5693</td>
<td>root</td>
<td>20</td>
<td>0</td>
<td>3448m</td>
<td>1368</td>
<td>912</td>
<td>R</td>
<td>7</td>
<td>0.0</td>
<td>0:00.07</td>
<td>top</td>
</tr>
<tr>
<td>17546</td>
<td>root</td>
<td>20</td>
<td>0</td>
<td>2044m</td>
<td>244m</td>
<td>79m</td>
<td>S</td>
<td>7</td>
<td>6.3</td>
<td>186:49:08</td>
<td>fed main event</td>
</tr>
<tr>
<td>18662</td>
<td>root</td>
<td>20</td>
<td>0</td>
<td>1806m</td>
<td>678m</td>
<td>263m</td>
<td>S</td>
<td>5</td>
<td>17.5</td>
<td>215:32:38</td>
<td>linux_iosd-img</td>
</tr>
<tr>
<td>30276</td>
<td>root</td>
<td>20</td>
<td>0</td>
<td>171m</td>
<td>42m</td>
<td>33m</td>
<td>S</td>
<td>5</td>
<td>1.1</td>
<td>125:06:77</td>
<td>repm</td>
</tr>
<tr>
<td>17835</td>
<td>root</td>
<td>20</td>
<td>0</td>
<td>935m</td>
<td>74m</td>
<td>63m</td>
<td>S</td>
<td>4</td>
<td>1.9</td>
<td>82:28:31</td>
<td>sif_mgr</td>
</tr>
<tr>
<td>18534</td>
<td>root</td>
<td>20</td>
<td>0</td>
<td>182m</td>
<td>150m</td>
<td>10m</td>
<td>S</td>
<td>2</td>
<td>3.9</td>
<td>8:12:08</td>
<td>smand</td>
</tr>
<tr>
<td>1</td>
<td>root</td>
<td>20</td>
<td>0</td>
<td>8440</td>
<td>4740</td>
<td>2184</td>
<td>S</td>
<td>0</td>
<td>0.1</td>
<td>0:09:52</td>
<td>systemd</td>
</tr>
<tr>
<td>2</td>
<td>root</td>
<td>20</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>S</td>
<td>0</td>
<td>0.0</td>
<td>0:00.00</td>
<td>kthread</td>
</tr>
<tr>
<td>3</td>
<td>root</td>
<td>20</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>S</td>
<td>0</td>
<td>0.0</td>
<td>0:02.86</td>
<td>ksoftirqd/0</td>
</tr>
<tr>
<td>5</td>
<td>root</td>
<td>0</td>
<td>-20</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>S</td>
<td>0</td>
<td>0.0</td>
<td>0:00.00</td>
<td>kworker/0:0H</td>
</tr>
<tr>
<td>7</td>
<td>root</td>
<td>RT</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>S</td>
<td>0</td>
<td>0.0</td>
<td>0:01.44</td>
<td>migration/0</td>
</tr>
<tr>
<td>8</td>
<td>root</td>
<td>20</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>S</td>
<td>0</td>
<td>0.0</td>
<td>0:00.00</td>
<td>rcu_bh</td>
</tr>
<tr>
<td>9</td>
<td>root</td>
<td>20</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>S</td>
<td>0</td>
<td>0.0</td>
<td>0:23:08</td>
<td>migration/0</td>
</tr>
<tr>
<td>10</td>
<td>root</td>
<td>20</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>S</td>
<td>0</td>
<td>0.0</td>
<td>0:58:04</td>
<td>rcuc/0</td>
</tr>
<tr>
<td>11</td>
<td>root</td>
<td>20</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>S</td>
<td>0</td>
<td>0.0</td>
<td>21:35:60</td>
<td>rcuc/1</td>
</tr>
<tr>
<td>12</td>
<td>root</td>
<td>RT</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>S</td>
<td>0</td>
<td>0.0</td>
<td>0:01:33</td>
<td>migration/1</td>
</tr>
</tbody>
</table>

Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>show processes cpu platform monitor location</td>
<td>Displays information about the CPU utilization of the IOS-XE processes.</td>
</tr>
</tbody>
</table>
show platform software status control-processor

To display platform software control-processor status, use the `show platform software status control-processor` command in privileged EXEC mode.

```plaintext
show platform software status control-processor
```

**Syntax Description**

- **brief** (Optional) Displays a summary of the platform control-processor status.

**Command Modes**

- Privileged EXEC (#)

**Command History**

- **Modification**
  - **Release**
  - Cisco IOS XE Denali 16.1.1  This command was introduced.

**Usage Guidelines**

Prior to Cisco IOS XE Denali 16.3.1, the Free Memory displayed in the command output was obtained from the underlying Linux kernel. This value was not accurate because some memory chunks that was available for use was not considered as free memory.

In Cisco IOS XE Denali 16.3.1, the free memory is accurately computed and displayed in the Free Memory field of the command output.

**Examples**

The following is sample output from the `show platform memory software status control-processor` command:

```plaintext
Switch# show platform software status control-processor

2-RP0: online, statistics updated 7 seconds ago
Load Average: healthy
  1-Min: 1.00, status: healthy, under 5.00
  5-Min: 1.21, status: healthy, under 5.00
  15-Min: 0.90, status: healthy, under 5.00
Memory (kb): healthy
  Total: 3976852
  Used: 2766284 (70%), status: healthy
  Free: 1210568 (30%), under 95%
Per-core Statistics
CPU0: CPU Utilization (percentage of time spent)
  User: 4.40, System: 1.70, Nice: 0.00, Idle: 93.80
  IRQ: 0.00, SIRQ: 0.10, IOwait: 0.00
CPU1: CPU Utilization (percentage of time spent)
  User: 3.80, System: 1.20, Nice: 0.00, Idle: 94.90
  IRQ: 0.00, SIRQ: 0.10, IOwait: 0.00
CPU2: CPU Utilization (percentage of time spent)
  User: 7.00, System: 1.10, Nice: 0.00, Idle: 91.89
  IRQ: 0.00, SIRQ: 0.00, IOwait: 0.00
CPU3: CPU Utilization (percentage of time spent)
  User: 4.49, System: 0.69, Nice: 0.00, Idle: 94.80
  IRQ: 0.00, SIRQ: 0.00, IOwait: 0.00

3-RP0: unknown, statistics updated 2 seconds ago
Load Average: healthy
  1-Min: 0.24, status: healthy, under 5.00
  5-Min: 0.27, status: healthy, under 5.00
```
15-Min: 0.32, status: healthy, under 5.00
Memory (kb): healthy
  Total: 3976852
  Used: 2706768 (68%), status: healthy
  Free: 1270084 (32%)
  Committed: 3299332 (83%), under 95%

Per-core Statistics
CPU0: CPU Utilization (percentage of time spent)
  User: 4.50, System: 1.20, Nice: 0.00, Idle: 94.20
  IRQ: 0.00, SIRQ: 0.10, IOWait: 0.00

CPU1: CPU Utilization (percentage of time spent)
  User: 5.20, System: 0.50, Nice: 0.00, Idle: 94.29
  IRQ: 0.00, SIRQ: 0.00, IOWait: 0.00

CPU2: CPU Utilization (percentage of time spent)
  User: 3.60, System: 0.70, Nice: 0.00, Idle: 95.69
  IRQ: 0.00, SIRQ: 0.00, IOWait: 0.00

CPU3: CPU Utilization (percentage of time spent)
  User: 3.00, System: 0.60, Nice: 0.00, Idle: 96.39
  IRQ: 0.00, SIRQ: 0.00, IOWait: 0.00

4-RP0: unknown, statistics updated 2 seconds ago
Load Average: healthy
  1-Min: 0.21, status: healthy, under 5.00
  5-Min: 0.24, status: healthy, under 5.00
  15-Min: 0.24, status: healthy, under 5.00
Memory (kb): healthy
  Total: 3976852
  Used: 1452404 (37%), status: healthy
  Free: 2524448 (63%)
  Committed: 1675120 (42%), under 95%

Per-core Statistics
CPU0: CPU Utilization (percentage of time spent)
  User: 2.30, System: 0.40, Nice: 0.00, Idle: 97.30
  IRQ: 0.00, SIRQ: 0.00, IOWait: 0.00

CPU1: CPU Utilization (percentage of time spent)
  User: 4.19, System: 0.69, Nice: 0.00, Idle: 95.10
  IRQ: 0.00, SIRQ: 0.00, IOWait: 0.00

CPU2: CPU Utilization (percentage of time spent)
  User: 4.79, System: 0.79, Nice: 0.00, Idle: 94.40
  IRQ: 0.00, SIRQ: 0.00, IOWait: 0.00

CPU3: CPU Utilization (percentage of time spent)
  User: 2.10, System: 0.40, Nice: 0.00, Idle: 97.50
  IRQ: 0.00, SIRQ: 0.00, IOWait: 0.00

9-RP0: unknown, statistics updated 4 seconds ago
Load Average: healthy
  1-Min: 0.20, status: healthy, under 5.00
  5-Min: 0.35, status: healthy, under 5.00
  15-Min: 0.35, status: healthy, under 5.00
Memory (kb): healthy
  Total: 3976852
  Used: 1451328 (36%), status: healthy
  Free: 2525524 (64%)
  Committed: 1675932 (42%), under 95%

Per-core Statistics
CPU0: CPU Utilization (percentage of time spent)
  User: 1.90, System: 0.50, Nice: 0.00, Idle: 97.60
  IRQ: 0.00, SIRQ: 0.00, IOWait: 0.00

CPU1: CPU Utilization (percentage of time spent)
  User: 4.39, System: 0.19, Nice: 0.00, Idle: 95.40
  IRQ: 0.00, SIRQ: 0.00, IOWait: 0.00

CPU2: CPU Utilization (percentage of time spent)
  User: 5.70, System: 1.00, Nice: 0.00, Idle: 93.30
  IRQ: 0.00, SIRQ: 0.00, IOWait: 0.00
CPU3: CPU Utilization (percentage of time spent)
  User: 1.30, System: 0.60, Nice: 0.00, Idle: 98.00
  IRQ: 0.00, SIRQ: 0.10, IOwait: 0.00

The following is sample output from the `show platform memory software status control-processor brief` command:

```
Switch# show platform software status control-processor brief

Load Average
Slot Status 1-Min 5-Min 15-Min
2-RP0 Healthy 1.10 1.21 0.91
3-RP0 Healthy 0.23 0.27 0.31
4-RP0 Healthy 0.11 0.21 0.22
9-RP0 Healthy 0.10 0.30 0.34

Memory (kB)
Slot Status Total Used (Pct) Free (Pct) Committed (Pct)
2-RP0 Healthy 3976852 2766956 (70%) 1209896 (30%) 3358352 (84%)
3-RP0 Healthy 3976852 2706824 (68%) 1270028 (32%) 3299276 (83%)
4-RP0 Healthy 3976852 1451888 (37%) 2524964 (63%) 1675076 (42%)
9-RP0 Healthy 3976852 1451580 (37%) 2525272 (63%) 1675952 (42%)

CPU Utilization
Slot CPU User System Nice Idle IRQ SIRQ IOwait
2-RP0 0 4.10 2.00 0.00 93.80 0.00 0.10 0.00
  1 4.60 1.00 0.00 94.30 0.00 0.10 0.00
  2 6.50 1.10 0.00 92.40 0.00 0.00 0.00
  3 5.59 1.19 0.00 93.20 0.00 0.00 0.00
3-RP0 0 2.80 1.20 0.00 95.90 0.00 0.10 0.00
  1 4.49 1.29 0.00 94.20 0.00 0.00 0.00
  2 5.30 1.60 0.00 93.10 0.00 0.00 0.00
  3 5.80 1.20 0.00 93.00 0.00 0.00 0.00
4-RP0 0 1.30 0.80 0.00 97.89 0.00 0.00 0.00
  1 1.30 0.20 0.00 98.50 0.00 0.00 0.00
  2 5.60 0.80 0.00 93.59 0.00 0.00 0.00
  3 5.09 0.19 0.00 94.70 0.00 0.00 0.00
9-RP0 0 3.99 0.69 0.00 95.30 0.00 0.00 0.00
  1 2.60 0.70 0.00 96.70 0.00 0.00 0.00
  2 4.49 0.89 0.00 94.60 0.00 0.00 0.00
  3 2.60 0.20 0.00 97.20 0.00 0.00 0.00
```
show platform software thread list

To display the list of threads on a platform, use the `show platform software thread list` command in privileged EXEC mode.

```
show platform software thread list switch { switch-number | active | standby } { 0 | F0 | FP active | R0 } pname { cdman | vidman | all } tname { main | pktio | rt | all }
```

### Syntax Description

- **switch switch-number**
  - Displays information about the switch. Enter the switch number.

- **active**
  - Specifies the active instance of the device.

- **standby**
  - Specifies standby instance of the device.

- **0**
  - Specifies the Shared Port Adapter (SPA) Interface Processor slot 0.

- **F0**
  - Specifies the Embedded Service Processor (ESP) slot 0.

- **FP active**
  - Specifies the active instance of Embedded Service Processor (ESP).

- **R0**
  - Specifies the Route Processor (RP) slot 0.

- **pname**
  - Specifies a process name. The possible values are `cdman`, `vidman`, and `all`.

- **tname**
  - Specifies a thread name. The possible values are `main`, `pktio`, `rt`, and `all`.

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE Gibraltar 16.11.1</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

### Command Modes

Privileged EXEC(#)

### Examples:

The following is sample output from the `show platform software thread list switch active R0 pname cdman tname all` command:

```
Device# show platform software thread list switch active R0 pname cdman tname all

Name   Tid  PPid  Group Id  Core  Vcswch  Nvcswch Status  Priority
------- -------- -------- -------- ------- -------- ---------- ----------
cdman   8407    7295    8407      1       0         0 S         20

Device# show platform software thread list switch active R0 pname cdman tname all

Name   Tid  PPid  Group Id  Core  Vcswch  Nvcswch Status  Priority
------- -------- -------- -------- ------- -------- ---------- ----------
cdman   8407    7295    8407      1       0         0 S         20
```

The table below describes the significant fields shown in the displays.
### Table 9: show platform software thread list Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Displays the command name associated with the process. Different threads in the same process may have different command values.</td>
</tr>
<tr>
<td>Tid</td>
<td>Displays the process ID.</td>
</tr>
<tr>
<td>PPid</td>
<td>Displays the process ID of the parent process.</td>
</tr>
<tr>
<td>Group Id</td>
<td>Displays the group ID.</td>
</tr>
<tr>
<td>Core</td>
<td>Displays processor information.</td>
</tr>
<tr>
<td>Vcswh</td>
<td>Displays the number of voluntary context switches.</td>
</tr>
<tr>
<td>Nvcswh</td>
<td>Displays the number of non-voluntary context switches.</td>
</tr>
<tr>
<td>Status</td>
<td>Displays the process status in human readable form.</td>
</tr>
<tr>
<td>Priority</td>
<td>Displays the negated scheduling priority.</td>
</tr>
<tr>
<td>TIME+</td>
<td>Displays the time since the start of the process.</td>
</tr>
<tr>
<td>Size</td>
<td>Displays the Resident Set Size (in kilobytes (KB)) that shows how much memory is allocated to that process in the RAM.</td>
</tr>
</tbody>
</table>
show processes cpu platform

To display information about the CPU utilization of the IOS-XE processes, use the **show processes cpu platform** command in privileged EXEC mode.

**show processes cpu platform** \[ [ sorted [ 1min | 5min | 5sec ] ] location switch \{ switch-number | active | standby \} \{ F0 | FP active | R0 | RP active \} \]

**Syntax Description**

- **sorted** (Optional) Displays output sorted based on percentage of CPU usage on a platform.
- **1min** (Optional) Sorts based on 1 minute intervals.
- **5min** (Optional) Sorts based on 5 minute intervals.
- **5sec** (Optional) Sorts based on 5 second intervals.
- **location** Specifies the Field Replaceable Unit (FRU) location.
- **switch**\{ switch-number | active | standby \} Displays information about the switch. Enter the switch number.
- **F0** Specifies the Embedded Service Processor (ESP) slot 0.
- **FP active** Specifies active instances on the Embedded Service Processor (ESP).
- **R0** Specifies the Route Processor (RP) slot 0.
- **RP active** Specifies active instances on the Route Processor (RP).

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE Gibraltar 16.11.1</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Command Modes**

Privileged EXEC (#)

**Examples:**

The following is sample output from the **show processes cpu platform** command:

```
Device# show processes cpu platform

CPU utilization for five seconds: 1%, one minute: 3%, five minutes: 2%
Core 0: CPU utilization for five seconds: 2%, one minute: 2%, five minutes: 2%
Core 1: CPU utilization for five seconds: 2%, one minute: 1%, five minutes: 1%
Core 2: CPU utilization for five seconds: 3%, one minute: 1%, five minutes: 1%
Core 3: CPU utilization for five seconds: 2%, one minute: 5%, five minutes: 2%
Pid PPid 5Sec 1Min 5Min Status Size Name
-----------------------------------------------
 1   0  0%  0%  0%  S   4876 systemd
```

**Interface and Hardware Commands**

107
The following is sample output from the `show processes cpu platform` command:

```
Device# show processes cpu platform sorted 5min location switch 5 R0

CPU utilization for five seconds: 0%, one minute: 0%, five minutes: 0%
Core 0: CPU utilization for five seconds: 1%, one minute: 1%, five minutes: 1%
Core 1: CPU utilization for five seconds: 1%, one minute: 1%, five minutes: 1%
Core 2: CPU utilization for five seconds: 1%, one minute: 1%, five minutes: 1%
Core 3: CPU utilization for five seconds: 2%, one minute: 0%, five minutes: 0%
Core 4: CPU utilization for five seconds: 0%, one minute: 0%, five minutes: 0%
Core 5: CPU utilization for five seconds: 0%, one minute: 0%, five minutes: 0%
Core 6: CPU utilization for five seconds: 0%, one minute: 0%, five minutes: 0%
Core 7: CPU utilization for five seconds: 0%, one minute: 0%, five minutes: 0%

<table>
<thead>
<tr>
<th>Pid</th>
<th>PPid</th>
<th>5Sec</th>
<th>1Min</th>
<th>5Min</th>
<th>Status</th>
<th>Size</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>16358</td>
<td>15516</td>
<td>4%</td>
<td>4%</td>
<td>4%</td>
<td>S</td>
<td>221376</td>
<td>fed main event</td>
</tr>
<tr>
<td>14062</td>
<td>12756</td>
<td>1%</td>
<td>1%</td>
<td>1%</td>
<td>S</td>
<td>52140</td>
<td>sif_mgr</td>
</tr>
<tr>
<td>32105</td>
<td>8618</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>S</td>
<td>260</td>
<td>inotifywait</td>
</tr>
<tr>
<td>31396</td>
<td>31393</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>S</td>
<td>36516</td>
<td>python2.7</td>
</tr>
<tr>
<td>31393</td>
<td>31271</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>S</td>
<td>2744</td>
<td>rdope.sh</td>
</tr>
<tr>
<td>31319</td>
<td>1</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>S</td>
<td>2648</td>
<td>rotee</td>
</tr>
<tr>
<td>31271</td>
<td>1</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>S</td>
<td>3852</td>
<td>pmn.sh</td>
</tr>
<tr>
<td>29671</td>
<td>2</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>S</td>
<td>1780</td>
<td>stack_sntp.sh</td>
</tr>
<tr>
<td>29341</td>
<td>29329</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>S</td>
<td>2788</td>
<td>stack_sntp.sh</td>
</tr>
</tbody>
</table>
```

Interface and Hardware Commands
The following is sample output from the `show processes cpu platform location switch 7 R0` command:

```
Device# show processes cpu platform location switch 7 R0

CPU utilization for five seconds: 3%, one minute: 3%, five minutes: 3%  
Core 0: CPU utilization for five seconds: 1%, one minute: 5%, five minutes: 5%  
Core 1: CPU utilization for five seconds: 1%, one minute: 11%, five minutes: 5%  
Core 2: CPU utilization for five seconds: 22%, one minute: 7%, five minutes: 6%  
Core 3: CPU utilization for five seconds: 5%, one minute: 6%, five minutes: 6%  
Core 4: CPU utilization for five seconds: 0%, one minute: 0%, five minutes: 0%  
Core 5: CPU utilization for five seconds: 0%, one minute: 0%, five minutes: 0%  
Core 6: CPU utilization for five seconds: 0%, one minute: 0%, five minutes: 0%  
Core 7: CPU utilization for five seconds: 0%, one minute: 0%, five minutes: 6%  

Pid  PPid  5Sec  1Min  5Min  Status  Size   Name  
-----------------------------------------------  
1    0     0%    0%    0%    S       8044 systemd
2    0     0%    0%    0%    S       0   kthreadd
```

...
show processes cpu platform history

To display information about the CPU usage history of a system, use the `show processes cpu platform history` command.

```
show processes cpu platform history [1min | 5min | 5sec | 60min] location
switch [switch-number | active | standby] | {0 | F0 | FP active | R0}
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>1min</code></td>
<td>(Optional) Displays CPU utilization history with 1 minute intervals.</td>
</tr>
<tr>
<td><code>5min</code></td>
<td>(Optional) Displays CPU utilization history with 5 minute intervals.</td>
</tr>
<tr>
<td><code>5sec</code></td>
<td>(Optional) Displays CPU utilization history with 5 second intervals.</td>
</tr>
<tr>
<td><code>60min</code></td>
<td>(Optional) Displays CPU utilization history with 60 minute intervals.</td>
</tr>
<tr>
<td><code>location</code></td>
<td>Specifies the Field Replaceable Unit (FRU) location.</td>
</tr>
<tr>
<td><code>switch switch-number</code></td>
<td>Displays information about the switch. Enter the switch number.</td>
</tr>
<tr>
<td><code>active</code></td>
<td>Specifies the active instance of the device.</td>
</tr>
<tr>
<td><code>standby</code></td>
<td>Specifies the standby instance of the device.</td>
</tr>
<tr>
<td><code>0</code></td>
<td>Specifies the Shared Port Adapter (SPA) Interface Processor slot 0.</td>
</tr>
<tr>
<td><code>F0</code></td>
<td>Specifies the Embedded Service Processor (ESP) slot 0.</td>
</tr>
<tr>
<td><code>FP active</code></td>
<td>Specifies active instances on the Embedded Service Processor (ESP).</td>
</tr>
<tr>
<td><code>R0</code></td>
<td>Specifies the Route Processor (RP) slot 0.</td>
</tr>
</tbody>
</table>

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE Gibraltar 16.11.1</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Command Modes**

Privileged EXEC (#)

**Examples:**

The following is sample output from the `show processes cpu platform` command:

```
Device# show processes cpu platform
```
CPU utilization for five seconds: 1%, one minute: 3%, five minutes: 2%
Core 0: CPU utilization for five seconds: 2%, one minute: 2%, five minutes: 2%
Core 1: CPU utilization for five seconds: 2%, one minute: 1%, five minutes: 1%
Core 2: CPU utilization for five seconds: 3%, one minute: 1%, five minutes: 1%
Core 3: CPU utilization for five seconds: 2%, one minute: 5%, five minutes: 2%

<table>
<thead>
<tr>
<th>Pid</th>
<th>PPid</th>
<th>5Sec</th>
<th>1Min</th>
<th>5Min</th>
<th>Status</th>
<th>Size</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>S</td>
<td>4876</td>
<td>systemd</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>S</td>
<td>0</td>
<td>kthreadd</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>S</td>
<td>0</td>
<td>ksoftirqd/0</td>
</tr>
<tr>
<td>5</td>
<td>2</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>S</td>
<td>0</td>
<td>kworker/0:0H</td>
</tr>
<tr>
<td>7</td>
<td>2</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>S</td>
<td>0</td>
<td>rcu_sched</td>
</tr>
<tr>
<td>8</td>
<td>2</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>S</td>
<td>0</td>
<td>rcu_bh</td>
</tr>
<tr>
<td>9</td>
<td>2</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>S</td>
<td>0</td>
<td>migration/0</td>
</tr>
<tr>
<td>10</td>
<td>2</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>S</td>
<td>0</td>
<td>watchdog/0</td>
</tr>
<tr>
<td>11</td>
<td>2</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>S</td>
<td>0</td>
<td>watchdog/1</td>
</tr>
<tr>
<td>12</td>
<td>2</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>S</td>
<td>0</td>
<td>migration/1</td>
</tr>
<tr>
<td>13</td>
<td>2</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>S</td>
<td>0</td>
<td>ksoftirqd/1</td>
</tr>
<tr>
<td>15</td>
<td>2</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>S</td>
<td>0</td>
<td>kworker/1:0H</td>
</tr>
<tr>
<td>16</td>
<td>2</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>S</td>
<td>0</td>
<td>watchdog/2</td>
</tr>
<tr>
<td>17</td>
<td>2</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>S</td>
<td>0</td>
<td>migration/2</td>
</tr>
<tr>
<td>18</td>
<td>2</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>S</td>
<td>0</td>
<td>ksoftirqd/2</td>
</tr>
<tr>
<td>20</td>
<td>2</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>S</td>
<td>0</td>
<td>kworker/2:0H</td>
</tr>
<tr>
<td>21</td>
<td>2</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>S</td>
<td>0</td>
<td>watchdog/3</td>
</tr>
<tr>
<td>22</td>
<td>2</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>S</td>
<td>0</td>
<td>migration/3</td>
</tr>
<tr>
<td>23</td>
<td>2</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>S</td>
<td>0</td>
<td>ksoftirqd/3</td>
</tr>
<tr>
<td>24</td>
<td>2</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>S</td>
<td>0</td>
<td>kworker/3:0</td>
</tr>
<tr>
<td>25</td>
<td>2</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>S</td>
<td>0</td>
<td>kworker/3:0H</td>
</tr>
<tr>
<td>26</td>
<td>2</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>S</td>
<td>0</td>
<td>kdevtmpfs</td>
</tr>
<tr>
<td>27</td>
<td>2</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>S</td>
<td>0</td>
<td>netns</td>
</tr>
<tr>
<td>28</td>
<td>2</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>S</td>
<td>0</td>
<td>perf</td>
</tr>
<tr>
<td>29</td>
<td>2</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>S</td>
<td>0</td>
<td>khungtaskd</td>
</tr>
<tr>
<td>30</td>
<td>2</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>S</td>
<td>0</td>
<td>writeback</td>
</tr>
<tr>
<td>31</td>
<td>2</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>S</td>
<td>0</td>
<td>kadm</td>
</tr>
<tr>
<td>32</td>
<td>2</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>S</td>
<td>0</td>
<td>hupgated</td>
</tr>
<tr>
<td>33</td>
<td>2</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>S</td>
<td>0</td>
<td>crypto</td>
</tr>
<tr>
<td>34</td>
<td>2</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>S</td>
<td>0</td>
<td>bioet</td>
</tr>
<tr>
<td>35</td>
<td>2</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>S</td>
<td>0</td>
<td>kbloctd</td>
</tr>
<tr>
<td>36</td>
<td>2</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>S</td>
<td>0</td>
<td>ata_sff</td>
</tr>
<tr>
<td>37</td>
<td>2</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>S</td>
<td>0</td>
<td>rpciod</td>
</tr>
<tr>
<td>63</td>
<td>2</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>S</td>
<td>0</td>
<td>kswapd0</td>
</tr>
<tr>
<td>64</td>
<td>2</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>S</td>
<td>0</td>
<td>vmstat</td>
</tr>
<tr>
<td>65</td>
<td>2</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>S</td>
<td>0</td>
<td>fsnotify_mark</td>
</tr>
</tbody>
</table>

The following is sample output from the `show processes cpu platform history 5sec` command:

```
Device# show processes cpu platform history 5sec

5 seconds ago, CPU utilization: 0%
10 seconds ago, CPU utilization: 0%
15 seconds ago, CPU utilization: 0%
20 seconds ago, CPU utilization: 0%
25 seconds ago, CPU utilization: 0%
30 seconds ago, CPU utilization: 0%
35 seconds ago, CPU utilization: 0%
40 seconds ago, CPU utilization: 0%
45 seconds ago, CPU utilization: 0%
50 seconds ago, CPU utilization: 0%
55 seconds ago, CPU utilization: 0%
60 seconds ago, CPU utilization: 0%
65 seconds ago, CPU utilization: 0%
70 seconds ago, CPU utilization: 0%
```
75 seconds ago, CPU utilization: 0%
80 seconds ago, CPU utilization: 0%
85 seconds ago, CPU utilization: 0%
90 seconds ago, CPU utilization: 0%
95 seconds ago, CPU utilization: 0%
100 seconds ago, CPU utilization: 0%
105 seconds ago, CPU utilization: 0%
110 seconds ago, CPU utilization: 0%
115 seconds ago, CPU utilization: 0%
120 seconds ago, CPU utilization: 0%
125 seconds ago, CPU utilization: 0%
130 seconds ago, CPU utilization: 0%
135 seconds ago, CPU utilization: 0%
140 seconds ago, CPU utilization: 0%
145 seconds ago, CPU utilization: 0%
150 seconds ago, CPU utilization: 0%
155 seconds ago, CPU utilization: 0%
160 seconds ago, CPU utilization: 0%
165 seconds ago, CPU utilization: 0%
170 seconds ago, CPU utilization: 0%
175 seconds ago, CPU utilization: 0%
180 seconds ago, CPU utilization: 0%
185 seconds ago, CPU utilization: 0%
190 seconds ago, CPU utilization: 0%
195 seconds ago, CPU utilization: 0%
200 seconds ago, CPU utilization: 0%
205 seconds ago, CPU utilization: 0%
210 seconds ago, CPU utilization: 0%
215 seconds ago, CPU utilization: 0%
220 seconds ago, CPU utilization: 0%
225 seconds ago, CPU utilization: 0%
230 seconds ago, CPU utilization: 0%
235 seconds ago, CPU utilization: 0%
240 seconds ago, CPU utilization: 0%
245 seconds ago, CPU utilization: 0%
250 seconds ago, CPU utilization: 0%

show processes cpu platform history
show processes cpu platform monitor

To display information about the CPU utilization of the IOS-XE processes, use the `show processes cpu platform monitor` command in privileged EXEC mode.

```
show processes cpu platform monitor location switch {switch-number | active | standby} {0 | F0 | R0}
```

**Syntax Description**

- `location` Displays information about the Field Replaceable Unit (FRU) location.
- `switch` Specifies the switch.
- `switch-number` Switch number.
- `active` Specifies the active instance.
- `standby` Specifies the standby instance.
- `0` Specifies the shared port adapter (SPA) interface processor slot 0.
- `F0` Specifies the Embedded Service Processor (ESP) slot 0.
- `R0` Specifies the Route Processor (RP) slot 0.

**Command Modes**

Privileged EXEC (#)

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE Denali 16.1.1</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

The output of the `show platform software process slot switch` and `show processes cpu platform monitor location` commands display the output of the Linux `top` command. The output of these commands display Free memory and Used memory as displayed by the Linux `top` command. The values displayed for the Free memory and Used memory by these commands do not match the values displayed by the output of other platform-memory related CLIs.

**Examples**

The following is sample output from the `show processes cpu monitor location switch active R0` command:

```
Switch# show processes cpu platform monitor location switch active R0

top - 00:04:21 up 1 day, 11:22, 0 users, load average: 0.42, 0.60, 0.78
Tasks: 312 total, 4 running, 308 sleeping, 0 stopped, 0 zombie
Cpu(s): 7.4%us, 3.3%sy, 0.0%ni, 89.2%id, 0.0%wa, 0.0%hi, 0.1%si, 0.0%st
Mem: 3976848k total, 3956928k used, 19916k free, 419312k buffers
Swap: 0k total, 0k used, 0k free, 1947036k cached

PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND
6294 root 20 0 3448 1368 912 R 9 0.0 00:00.07 top
17546 root 20 0 2044m 244m 79m S 7 6.3 187:02.07 fed main event
30276 root 20 0 171m 42m 33m S 7 1.1 125:15.54 repm
16 root 20 0 0 0 0 S 5 0.0 22:07.92 rcuc/2
21 root 20 0 0 0 0 R 5 0.0 22:13.24 rcuc/3
```
show processes cpu platform monitor

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>show platform software process slot switch</td>
<td>Displays platform software process switch information.</td>
</tr>
</tbody>
</table>
To display memory usage for each Cisco IOS XE process, use the `show processes memory platform` command in privileged EXEC mode.

```
show processes memory platform [ [ detailed { name process-name | process-id process-ID } [ location | maps [ location ] | smaps [ location ] ] | location | sorted { location } ] switch { switch-number | active | standby } [ 0 | F0 | R0 ] | accounting ]
```

### Syntax Description

- **accounting** (Optional) Displays the top memory allocators for each Cisco IOS XE process.
- **detailed** (Optional) Displays detailed memory information for a specified Cisco IOS XE process.
- **name process-name** (Optional) Displays the Cisco IOS XE process name. Enter the process name.
- **process-id process-ID** (Optional) Displays the Cisco IOS XE process ID. Enter the process ID.
- **location** (Optional) Displays information about the Field Replaceable Unit (FRU) location.
- **maps** (Optional) Displays memory maps of a process.
- **smaps** (Optional) Displays static memory maps of a process.
- **sorted** (Optional) Displays the sorted output based on the Resident Set Size (RSS) memory used by Cisco IOS XE process.
- **switch switch-number** Displays information about the device.
- **active** Displays information about the active instance of the device.
- **standby** Displays information about the standby instance of the device.
- **0** Displays information about Shared Port Adapter (SPA)-Inter-Processor slot 0.
- **F0** Displays information about Embedded Service Processor (ESP) slot 0.
- **R0** Displays information about Route Processor (RP) slot 0.

### Command Modes

Privileged EXEC (#)
Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE Gibraltar 16.11.1</td>
<td>This command was introduced. This command was modified. The keyword \textit{accounting} was added. The Total column was deleted from the output.</td>
</tr>
</tbody>
</table>

Examples

The following is a sample output from the \texttt{show processes memory platform} command:

```
device# show processes memory platform

System memory: 3976852K total, 2761580K used, 1215272K free,
Lowest: 1215272K

<table>
<thead>
<tr>
<th>Pid</th>
<th>Text</th>
<th>Data</th>
<th>Stack</th>
<th>Dynamic</th>
<th>RSS</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1246</td>
<td>4400</td>
<td>132</td>
<td>1308</td>
<td>4400</td>
<td>systemd</td>
</tr>
<tr>
<td>96</td>
<td>233</td>
<td>2796</td>
<td>132</td>
<td>132</td>
<td>2796</td>
<td>systemd-journal</td>
</tr>
<tr>
<td>105</td>
<td>284</td>
<td>1796</td>
<td>132</td>
<td>176</td>
<td>1796</td>
<td>systemd-udevd</td>
</tr>
<tr>
<td>707</td>
<td>52</td>
<td>2660</td>
<td>132</td>
<td>172</td>
<td>2660</td>
<td>in.telnetd</td>
</tr>
<tr>
<td>744</td>
<td>968</td>
<td>3264</td>
<td>132</td>
<td>1700</td>
<td>3264</td>
<td>brelay.sh</td>
</tr>
<tr>
<td>835</td>
<td>52</td>
<td>2660</td>
<td>132</td>
<td>172</td>
<td>2660</td>
<td>in.telnetd</td>
</tr>
<tr>
<td>863</td>
<td>968</td>
<td>3264</td>
<td>132</td>
<td>1700</td>
<td>3264</td>
<td>brelay.sh</td>
</tr>
<tr>
<td>928</td>
<td>968</td>
<td>3976</td>
<td>132</td>
<td>2312</td>
<td>3976</td>
<td>reflector.sh</td>
</tr>
<tr>
<td>933</td>
<td>968</td>
<td>3976</td>
<td>132</td>
<td>2312</td>
<td>3976</td>
<td>droputil.sh</td>
</tr>
<tr>
<td>934</td>
<td>968</td>
<td>2140</td>
<td>132</td>
<td>528</td>
<td>2140</td>
<td>oom.sh</td>
</tr>
<tr>
<td>936</td>
<td>173</td>
<td>936</td>
<td>132</td>
<td>132</td>
<td>936</td>
<td>xinetd</td>
</tr>
<tr>
<td>945</td>
<td>968</td>
<td>1472</td>
<td>132</td>
<td>132</td>
<td>1472</td>
<td>libvirtd.sh</td>
</tr>
<tr>
<td>947</td>
<td>592</td>
<td>43164</td>
<td>132</td>
<td>3096</td>
<td>43164</td>
<td>repm</td>
</tr>
<tr>
<td>954</td>
<td>45</td>
<td>932</td>
<td>132</td>
<td>132</td>
<td>932</td>
<td>rpcbind</td>
</tr>
<tr>
<td>986</td>
<td>482</td>
<td>3476</td>
<td>132</td>
<td>132</td>
<td>3476</td>
<td>libvirtd</td>
</tr>
<tr>
<td>988</td>
<td>66</td>
<td>940</td>
<td>132</td>
<td>132</td>
<td>940</td>
<td>rpc.statd</td>
</tr>
<tr>
<td>993</td>
<td>968</td>
<td>928</td>
<td>132</td>
<td>132</td>
<td>928</td>
<td>boothelper_evt.</td>
</tr>
<tr>
<td>1017</td>
<td>23</td>
<td>640</td>
<td>132</td>
<td>132</td>
<td>640</td>
<td>notifywait</td>
</tr>
<tr>
<td>1089</td>
<td>102</td>
<td>1200</td>
<td>132</td>
<td>132</td>
<td>1200</td>
<td>rpc.mountd</td>
</tr>
<tr>
<td>1328</td>
<td>9</td>
<td>2940</td>
<td>132</td>
<td>148</td>
<td>2940</td>
<td>rotee</td>
</tr>
<tr>
<td>1353</td>
<td>39</td>
<td>532</td>
<td>132</td>
<td>132</td>
<td>532</td>
<td>sleep</td>
</tr>
</tbody>
</table>
```

The following is a sample output from the \texttt{show processes memory platform accounting} command:

```
device# show processes memory platform accounting

Hourly Stats

<table>
<thead>
<tr>
<th>process</th>
<th>callsite_ID(bytes)</th>
<th>max_diff_bytes</th>
<th>callsite_ID(calls)</th>
<th>max_diff_calls</th>
<th>tracekey</th>
<th>timestamp(UTC)</th>
</tr>
</thead>
<tbody>
<tr>
<td>smand_rp_0</td>
<td>3624155137</td>
<td>172389</td>
<td>3624155138</td>
<td>50</td>
<td>#3e0e4361082c702e5bf1afbd90e6313</td>
<td>2018-09-04 14:23</td>
</tr>
<tr>
<td>linux_iosd-img_rp_0</td>
<td>3626295305</td>
<td>49188</td>
<td>3624155138</td>
<td>12</td>
<td>#545420bd869d25eb5ab826182e5d9ce</td>
<td>2018-09-04 12:03</td>
</tr>
<tr>
<td>btman_rp_0</td>
<td>3624737792</td>
<td>17080</td>
<td>2953915394</td>
<td>64</td>
<td>#6888bd956a3c4fcf049e31ba07a036</td>
<td>2018-09-04 22:29</td>
</tr>
</tbody>
</table>
```
The following is a sample output from the `show processes memory platform sorted` command:

```
7885 149848 684864 136 80 684864 linux_iosd-imag
9655 3787 264964 136 18004 264964 wcm
```

Interface and Hardware Commands

```
show processes memory platform sorted
```

System memory: 3976852K total, 2762884K used, 1213968K free,
Lowest: 1213968K

<table>
<thead>
<tr>
<th>Pid</th>
<th>Text</th>
<th>Data</th>
<th>Stack</th>
<th>Dynamic</th>
<th>RSS</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>7885</td>
<td>149848</td>
<td>684864</td>
<td>136</td>
<td>80</td>
<td>684864</td>
<td>linux_iosd-imag</td>
</tr>
<tr>
<td>9655</td>
<td>3787</td>
<td>264964</td>
<td>136</td>
<td>18004</td>
<td>264964</td>
<td>wcm</td>
</tr>
</tbody>
</table>
The following is sample output from the `show processes memory platform sorted location switch active R0` command:

device# show processes memory platform sorted location switch active R0
System memory: 3976852K total, 2762884K used, 1213968K free,
Lowest: 1213968K

<table>
<thead>
<tr>
<th>Pid</th>
<th>Text</th>
<th>Data</th>
<th>Stack</th>
<th>Dynamic</th>
<th>RSS</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>7885</td>
<td>149848</td>
<td>684864</td>
<td>136</td>
<td>80</td>
<td>684864</td>
<td>linux_iosd-imag</td>
</tr>
<tr>
<td>9655</td>
<td>3787</td>
<td>264964</td>
<td>136</td>
<td>18004</td>
<td>264964</td>
<td>wcm</td>
</tr>
<tr>
<td>17261</td>
<td>324</td>
<td>248588</td>
<td>132</td>
<td>103908</td>
<td>248588</td>
<td>fed main event</td>
</tr>
<tr>
<td>4268</td>
<td>391</td>
<td>102084</td>
<td>136</td>
<td>5596</td>
<td>102084</td>
<td>cli_agent</td>
</tr>
<tr>
<td>4856</td>
<td>357</td>
<td>93388</td>
<td>132</td>
<td>3680</td>
<td>93388</td>
<td>dbm</td>
</tr>
<tr>
<td>17067</td>
<td>1087</td>
<td>77912</td>
<td>136</td>
<td>1796</td>
<td>77912</td>
<td>platform_mgr</td>
</tr>
</tbody>
</table>
show processes platform

To display information about the IOS-XE processes running on a platform, use the `show processes platform` command in privileged EXEC mode.

```
show processes platform [detailed name process-name] [location switch [switch-number | active | standby] {0 | F0 | FP active | R0}]
```

- **detailed** (Optional) Displays detailed information of the specified IOS-XE process.
- **name process-name** (Optional) Specifies the process name.
- **location** (Optional) Specifies the Field Replaceable Unit (FRU) location.
- **switch switch-number** (Optional) Displays information about the switch.
- **active** (Optional) Specifies the active instance of the device.
- **standby** (Optional) Specifies standby instance of the device.
- **0** Specifies the Shared Port Adapter (SPA) Interface Processor slot 0.
- **F0** Specifies the Embedded Service Processor (ESP) slot 0.
- **FP active** Specifies the active instance in the Embedded Service Processor (ESP).
- **R0** Specifies the Route Processor (RP) slot 0.

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE Gibraltar 16.11.1</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Command Modes**

Privileged EXEC(#)  

**Examples:**

The following is sample output from the `show processes platform` command:

```
Device# show processes platform

CPU utilization for five seconds: 1%, one minute: 2%, five minutes: 1%

<table>
<thead>
<tr>
<th>Pid</th>
<th>PPid</th>
<th>Status</th>
<th>Size</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>S</td>
<td>4876</td>
<td>systemd</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>S</td>
<td>0</td>
<td>kthreadd</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>S</td>
<td>0</td>
<td>ksoftirqd/0</td>
</tr>
<tr>
<td>5</td>
<td>2</td>
<td>S</td>
<td>0</td>
<td>kworker/0:0H</td>
</tr>
<tr>
<td>7</td>
<td>2</td>
<td>S</td>
<td>0</td>
<td>rcu_sched</td>
</tr>
<tr>
<td>8</td>
<td>2</td>
<td>S</td>
<td>0</td>
<td>rcu_bh</td>
</tr>
<tr>
<td>9</td>
<td>2</td>
<td>S</td>
<td>0</td>
<td>migration/0</td>
</tr>
<tr>
<td>10</td>
<td>2</td>
<td>S</td>
<td>0</td>
<td>watchdog/0</td>
</tr>
<tr>
<td>11</td>
<td>2</td>
<td>S</td>
<td>0</td>
<td>watchdog/1</td>
</tr>
<tr>
<td>12</td>
<td>2</td>
<td>S</td>
<td>0</td>
<td>migration/1</td>
</tr>
</tbody>
</table>
```
The table below describes the significant fields shown in the displays.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pid</td>
<td>Displays the process ID.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Ppid</td>
<td>Displays the process ID of the parent process.</td>
</tr>
<tr>
<td>Status</td>
<td>Displays the process status in human readable form.</td>
</tr>
<tr>
<td>Size</td>
<td>Displays the Resident Set Size (in kilobytes (KB)) that shows how much memory is allocated to that process in the RAM.</td>
</tr>
<tr>
<td>Name</td>
<td>Displays the command name associated with the process. Different threads in the same process may have different command values.</td>
</tr>
</tbody>
</table>
show power inline

To display the Power over Ethernet (PoE) status for the specified PoE port, the specified stack member, or for all PoE ports in the switch stack, use the `show power inline` command in EXEC mode.

```
show power inline   [{police | priority}]   [{interface-id | module stack-member-number}]   [detail]
```

**Syntax Description**

- `police`  
  (Optional) Displays the power policing information about real-time power consumption.

- `priority`  
  (Optional) Displays the power inline port priority for each port.

- `interface-id`  
  (Optional) ID of the physical interface.

- `module stack-member-number`  
  (Optional) Limits the display to ports on the specified stack member.  
  The range is 1 to 9.  
  This keyword is supported only on stacking-capable switches.

- `detail`  
  (Optional) Displays detailed output of the interface or module.

**Command Modes**

- User EXEC
- Privileged EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE 3.2SE</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Examples**

This is an example of output from the `show power inline` command. The table that follows describes the output fields.

```
Device> show power inline
Module  Available  Used  Remaining
       (Watts)    (Watts)  (Watts)
-------  ---------  ------  -------
1        n/a      n/a   n/a
2        n/a      n/a   n/a
3        1440.0   15.4  1424.6
4        720.0    6.3   713.7

Interface Admin  Oper  Power  Device  Class Max
               (Watts)   -------  --------  ----    -----  -----  ------
               ---------  ---------  ------  --------  -------
G13/0/1       auto  off      0.0   n/a       n/a   30.0  30.0
G13/0/2       auto  off      0.0   n/a       n/a   30.0  30.0
G13/0/3       auto  off      0.0   n/a       n/a   30.0  30.0
G13/0/4       auto  off      0.0   n/a       n/a   30.0  30.0
G13/0/5       auto  off      0.0   n/a       n/a   30.0  30.0
G13/0/6       auto  off      0.0   n/a       n/a   30.0  30.0
G13/0/7       auto  off      0.0   n/a       n/a   30.0  30.0
G13/0/8       auto  off      0.0   n/a       n/a   30.0  30.0
```
This is an example of output from the `show power inline interface-id` command on a switch port:

```
Device> show power inline gigabitethernet1/0/1
```

<table>
<thead>
<tr>
<th>Interface</th>
<th>Admin</th>
<th>Oper</th>
<th>Power</th>
<th>Device</th>
<th>Class</th>
<th>Max (Watts)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gi1/0/1</td>
<td>auto</td>
<td>off</td>
<td>0.0</td>
<td>n/a</td>
<td>n/a</td>
<td>30.0</td>
</tr>
</tbody>
</table>

This is an example of output from the `show power inline module switch-number` command on stack member 3. The table that follows describes the output fields.

```
Device> show power inline module 3
```

<table>
<thead>
<tr>
<th>Module</th>
<th>Available (Watts)</th>
<th>Used (Watts)</th>
<th>Remaining (Watts)</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>865.0</td>
<td>864.0</td>
<td>1.0</td>
</tr>
</tbody>
</table>

```
Interface Admin Oper Power Device Class Max (Watts)
--------- ------ ---------- ------- ------------------- ----- ----
Gi3/0/1   auto  power-deny 4.0   n/a   n/a   15.4
Gi3/0/2   auto  off          0.0   n/a   n/a   15.4
Gi3/0/3   auto  off          0.0   n/a   n/a   15.4
Gi3/0/4   auto  off          0.0   n/a   n/a   15.4
Gi3/0/5   auto  off          0.0   n/a   n/a   15.4
Gi3/0/6   auto  off          0.0   n/a   n/a   15.4
Gi3/0/7   auto  off          0.0   n/a   n/a   15.4
Gi3/0/8   auto  off          0.0   n/a   n/a   15.4
Gi3/0/9   auto  off          0.0   n/a   n/a   15.4
Gi3/0/10  auto  off          0.0   n/a   n/a   15.4
<output truncated>
```

Table 11: show power inline Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Available</td>
<td>The total amount of configured power(^1) on the PoE switch in watts (W).</td>
</tr>
<tr>
<td>Used</td>
<td>The amount of configured power that is allocated to PoE ports in watts.</td>
</tr>
<tr>
<td>Remaining</td>
<td>The amount of configured power in watts that is not allocated to ports in the system. (Available – Used = Remaining)</td>
</tr>
<tr>
<td>Admin</td>
<td>Administration mode: auto, off, static.</td>
</tr>
</tbody>
</table>
Field | Description
--- | ---
**Oper** | Operating mode:
- on—The powered device is detected, and power is applied.
- off—No PoE is applied.
- faulty—Device detection or a powered device is in a faulty state.
- power-deny—A powered device is detected, but no PoE is available, or the maximum wattage exceeds the detected powered-device maximum.

**Power** | The maximum amount of power that is allocated to the powered device in watts. This value is the same as the value in the Cutoff Power field in the show power inline police command output.

**Device** | The device type detected: n/a, unknown, Cisco powered-device, IEEE powered-device, or the name from CDP.

**Class** | The IEEE classification: n/a or a value from 0 to 4.

**Max** | The maximum amount of power allocated to the powered device in watts.

**AdminPowerMax** | The maximum amount power allocated to the powered device in watts when the switch polices the real-time power consumption. This value is the same as the Max field value.

**AdminConsumption** | The power consumption of the powered device in watts when the switch polices the real-time power consumption. If policing is disabled, this value is the same as the AdminPowerMax field value.

1 The configured power is the power that you manually specify or that the switch specifies by using CDP power negotiation or the IEEE classification, which is different than the real-time power that is monitored with the power sensing feature.

This is an example of output from the **show power inline police** command on a stacking-capable switch:

```
Device> show power inline police
Module Available Used Remaining
(Watts) (Watts) (Watts)
------- -------- -------- --------
1 370.0 0.0 370.0
3 865.0 864.0 1.0

Interface Admin Oper Admin Oper Cutoff Oper
--------- ------ ---------- ------ ---------- ------ -----
Gi1/0/1 auto off none n/a n/a 0.0
Gi1/0/2 auto off log n/a 5.4 0.0
Gi1/0/3 auto off errdisable n/a 5.4 0.0
Gi1/0/4 off off none n/a n/a 0.0
Gi1/0/5 off off log n/a 5.4 0.0
Gi1/0/6 off off errdisable n/a 5.4 0.0
Gi1/0/7 auto off none n/a n/a 0.0
Gi1/0/8 auto off log n/a 5.4 0.0
Gi1/0/9 auto on none n/a n/a 5.1
Gi1/0/10 auto on log ok 5.4 4.2
Gi1/0/11 auto on log log 5.4 5.9
Gi1/0/12 auto on errdisable ok 5.4 4.2
```
In the previous example:

- The Gi1/0/1 port is shut down, and policing is not configured.
- The Gi1/0/2 port is shut down, but policing is enabled with a policing action to generate a syslog message.
- The Gi1/0/3 port is shut down, but policing is enabled with a policing action is to shut down the port.
- Device detection is disabled on the Gi1/0/4 port, power is not applied to the port, and policing is disabled.
- Device detection is disabled on the Gi1/0/5 port, and power is not applied to the port, but policing is enabled with a policing action to generate a syslog message.
- Device detection is disabled on the Gi1/0/6 port, and power is not applied to the port, but policing is enabled with a policing action to shut down the port.
- The Gi1/0/7 port is up, and policing is disabled, but the switch does not apply power to the connected device.
- The Gi1/0/8 port is up, and policing is enabled with a policing action to generate a syslog message, but the switch does not apply power to the powered device.
- The Gi1/0/9 port is up and connected to a powered device, and policing is disabled.
- The Gi1/0/10 port is up and connected to a powered device, and policing is enabled with a policing action to generate a syslog message. The policing action does not take effect because the real-time power consumption is less than the cutoff value.
- The Gi1/0/11 port is up and connected to a powered device, and policing is enabled with a policing action to generate a syslog message.
- The Gi1/0/12 port is up and connected to a powered device, and policing is enabled with a policing action to shut down the port. The policing action does not take effect because the real-time power consumption is less than the cutoff value.
- The Gi1/0/13 port is up and connected to a powered device, and policing is enabled with a policing action to shut down the port.

This is an example of output from the `show power inline police interface-id` command on a standalone switch. The table that follows describes the output fields.
### Table 12: `show power inline police` Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Available</td>
<td>The total amount of configured power(^2) on the switch in watts (W).</td>
</tr>
<tr>
<td>Used</td>
<td>The amount of configured power allocated to PoE ports in watts.</td>
</tr>
<tr>
<td>Remaining</td>
<td>The amount of configured power in watts that is not allocated to ports in the system. (Available – Used = Remaining)</td>
</tr>
<tr>
<td>Admin State</td>
<td>Administration mode: auto, off, static.</td>
</tr>
</tbody>
</table>
| Oper State     | Operating mode:  
  - errdisable—Policing is enabled.  
  - faulty—Device detection on a powered device is in a faulty state.  
  - off—No PoE is applied.  
  - on—The powered device is detected, and power is applied.  
  - power-deny—A powered device is detected, but no PoE is available, or the real-time power consumption exceeds the maximum power allocation.  

**Note**  
The operating mode is the current PoE state for the specified PoE port, the specified stack member, or for all PoE ports on the switch. |
| Admin Police   | Status of the real-time power-consumption policing feature:  
  - errdisable—Policing is enabled, and the switch shuts down the port when the real-time power consumption exceeds the maximum power allocation.  
  - log—Policing is enabled, and the switch generates a syslog message when the real-time power consumption exceeds the maximum power allocation.  
  - none—Policing is disabled. |
| Oper Police    | Policing status:  
  - errdisable—The real-time power consumption exceeds the maximum power allocation, and the switch shuts down the PoE port.  
  - log—The real-time power consumption exceeds the maximum power allocation, and the switch generates a syslog message.  
  - n/a—Device detection is disabled, power is not applied to the PoE port, or no policing action is configured.  
  - ok—Real-time power consumption is less than the maximum power allocation. |
| Cutoff Power   | The maximum power allocated on the port. When the real-time power consumption is greater than this value, the switch takes the configured policing action. |
| Oper Power     | The real-time power consumption of the powered device. |

---

\(^2\) The configured power is the power that you manually specify or that the switch specifies by using CDP power negotiation or the IEEE classification, which is different than the real-time power that is monitored with the power sensing feature.

This is an example of output from the `show power inline priority` command on a standalone switch.
Device> **show power inline priority**

<table>
<thead>
<tr>
<th>Interface</th>
<th>Admin State</th>
<th>Oper State</th>
<th>Priority</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gi1/0/1</td>
<td>auto</td>
<td>off</td>
<td>low</td>
</tr>
<tr>
<td>Gi1/0/2</td>
<td>auto</td>
<td>off</td>
<td>low</td>
</tr>
<tr>
<td>Gi1/0/3</td>
<td>auto</td>
<td>off</td>
<td>low</td>
</tr>
<tr>
<td>Gi1/0/4</td>
<td>auto</td>
<td>off</td>
<td>low</td>
</tr>
<tr>
<td>Gi1/0/5</td>
<td>auto</td>
<td>off</td>
<td>low</td>
</tr>
<tr>
<td>Gi1/0/6</td>
<td>auto</td>
<td>off</td>
<td>low</td>
</tr>
<tr>
<td>Gi1/0/7</td>
<td>auto</td>
<td>off</td>
<td>low</td>
</tr>
<tr>
<td>Gi1/0/8</td>
<td>auto</td>
<td>off</td>
<td>low</td>
</tr>
<tr>
<td>Gi1/0/9</td>
<td>auto</td>
<td>off</td>
<td>low</td>
</tr>
</tbody>
</table>
show stack-power

To display information about StackPower stacks or switches in a power stack, use the `show stack-power` command in EXEC mode.

```
{show stack-power [{budgeting | detail | load-shedding | neighbors}] [order power-stack-name] [{stack-name [stack-id] | switch [switch-id]}]}
```

### Syntax Description

- **budgeting** *(Optional)* Displays the stack power budget table.
- **detail** *(Optional)* Displays the stack power stack details.
- **load-shedding** *(Optional)* Displays the stack power load shedding table.
- **neighbors** *(Optional)* Displays the stack power neighbor table.
- **order power-stack-name** *(Optional)* Displays the load shedding priority for a power stack.
  
  **Note** This keyword is available only after the `load-shedding` keyword.

- **stack-name** *(Optional)* Displays budget table, details, or neighbors for all power stacks or the specified power stack.
  
  **Note** This keyword is not available after the `load-shedding` keyword.

- **stack-id** *(Optional)* Power stack ID for the power stack. The stack ID must be 31 characters or less.

- **switch** *(Optional)* Displays budget table, details, load-shedding, or neighbors for all switches or the specified switch.

- **switch-id** *(Optional)* Switch ID for the switch. The switch number is from 1 to 9.

### Command Modes

Privileged EXEC

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE Denali 16.3.2</td>
<td>Support for all the options was enabled for this command.</td>
</tr>
<tr>
<td>Cisco IOS XE Denali 16.1.1</td>
<td>This command was reintroduced.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

This command is available only on switch stacks running the IP Base or IP Services image.

If a switch is shut down because of load shedding, the output of the `show stack-power` command still includes the MAC address of the shutdown neighbor switch. The command output shows the stack power topology even if there is not enough power to power a switch.

### Examples

This is an example of output from the `show stack-power` command:
### Device# `show stack-power`

<table>
<thead>
<tr>
<th>Name</th>
<th>Mode</th>
<th>Topology</th>
<th>Pwr (W)</th>
<th>Pwr (W)</th>
<th>Pwr (W)</th>
<th>Pwr (W)</th>
<th>SW</th>
<th>PS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Powerstack-1</td>
<td>SP-PS</td>
<td>Stndaln</td>
<td>350</td>
<td>150</td>
<td>200</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

This is an example of output from the `show stack-power` command:

### Device# `show stack-power budgeting`

<table>
<thead>
<tr>
<th>Name</th>
<th>Mode</th>
<th>Topology</th>
<th>Pwr (W)</th>
<th>Pwr (W)</th>
<th>Pwr (W)</th>
<th>Pwr (W)</th>
<th>SW</th>
<th>PS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Powerstack-1</td>
<td>SP-PS</td>
<td>Stndaln</td>
<td>350</td>
<td>150</td>
<td>200</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Power Stack</th>
<th>PS-A</th>
<th>PS-B</th>
<th>Power Budget</th>
<th>Alloc Power</th>
<th>Avail Pwr</th>
<th>Consumd Pwr</th>
<th>Sys/PoE (W)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Powerstack-1</td>
<td>350</td>
<td>0</td>
<td>200</td>
<td>200</td>
<td>0</td>
<td>60</td>
<td>0</td>
</tr>
</tbody>
</table>

Totals:

<p>| | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>200</td>
<td>0</td>
<td>60</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
show system mtu

To display the global maximum transmission unit (MTU) or maximum packet size set for the switch, use the show system mtu command in privileged EXEC mode.

show system mtu

Syntax Description
This command has no arguments or keywords.

Command Default
None

Command Modes
Privileged EXEC

Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE 3.2SE</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

Usage Guidelines
For information about the MTU values and the stack configurations that affect the MTU values, see the system mtu command.

Examples
This is an example of output from the show system mtu command:

Device# show system mtu
Global Ethernet MTU is 1500 bytes.
show tech-support

To automatically run `show` commands that display system information, use the `show tech-support` command in the privilege EXEC mode.

```
show tech-support
    [cef|cft|eigrp|evc|fnf|ipc|ipmulticast|ipsec|mfib|nat|nbar|onep|ospf|page|password|rsvp|subscriber|vrrp|wccp]
```

### Syntax Description

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>cef</td>
<td>(Optional) Displays CEF related information.</td>
</tr>
<tr>
<td>cft</td>
<td>(Optional) Displays CFT related information.</td>
</tr>
<tr>
<td>eigrp</td>
<td>(Optional) Displays EIGRP related information.</td>
</tr>
<tr>
<td>evc</td>
<td>(Optional) Displays EVC related information.</td>
</tr>
<tr>
<td>fnf</td>
<td>(Optional) Displays flexible netflow related information.</td>
</tr>
<tr>
<td>ipc</td>
<td>(Optional) Displays IPC related information.</td>
</tr>
<tr>
<td>ipmulticast</td>
<td>(Optional) Displays IP multicast related information.</td>
</tr>
<tr>
<td>ipsec</td>
<td>(Optional) Displays IPSEC related information.</td>
</tr>
<tr>
<td>mfib</td>
<td>(Optional) Displays MFIB related information.</td>
</tr>
<tr>
<td>nat</td>
<td>(Optional) Displays NAT related information.</td>
</tr>
<tr>
<td>onep</td>
<td>(Optional) Displays ONEP related information.</td>
</tr>
<tr>
<td>ospf</td>
<td>(Optional) Displays OSPF related information.</td>
</tr>
<tr>
<td>page</td>
<td>(Optional) Displays the command output on a single page at a time. Use the Return key to display the next line of output or use the space bar to display the next page of information. If not used, the output scrolls (that is, it does not stop for page breaks). Press the Ctrl-C keys to stop the command output.</td>
</tr>
<tr>
<td>password</td>
<td>(Optional) Leaves passwords and other security information in the output. If not used, passwords and other security-sensitive information in the output are replaced with the label &quot;&lt;removed&gt;&quot;.</td>
</tr>
<tr>
<td>subscriber</td>
<td>(Optional) Displays subscriber related information.</td>
</tr>
<tr>
<td>vrrp</td>
<td>(Optional) Displays VRRP related information.</td>
</tr>
<tr>
<td>wccp</td>
<td>(Optional) Displays WCCP related information.</td>
</tr>
</tbody>
</table>

### Command Modes

Privileged EXEC (#)
Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE Gibraltar 16.11.1</td>
<td>This command was enhanced to display the output of the show logging onboard uptime command</td>
</tr>
</tbody>
</table>
| Cisco IOS XE Denali 16.3.2 | This command was enhanced to display of the outputs of the following commands in the output modifier:  
  - show power inline  
  - show platform software  
  - show power inline police  
  - show stack-power budgeting |
| Cisco IOS XE Denali 16.1.1 | This command was implemented on the Cisco Catalyst 3850 Series Switches |

Usage Guidelines

The output of the `show logging onboard uptime` command is displayed in the output of the `show tech-support` command.

The output from the `show tech-support` command is very long. To better manage this output, you can redirect the output to a file (for example, `show tech-support > filename`) in the local writable storage file system or the remote file system. Redirecting the output to a file also makes sending the output to your Cisco Technical Assistance Center (TAC) representative easier.

You can use one of the following redirection methods:

- `>` `filename` - Redirects the output to a file.
- `>>` `filename` - Redirects the output to a file in append mode.
show tech-support diagnostic

To display diagnostic information for technical support, use the `show tech-support diagnostic` command in privileged EXEC mode.

```
show tech-support diagnostic
```

<table>
<thead>
<tr>
<th>Syntax Description</th>
<th>This command has no arguments or keywords.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Command Modes</td>
<td>Privileged EXEC (#)</td>
</tr>
</tbody>
</table>

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE Gibraltar 16.11.1</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

The output of this command is very long. To better manage this output, you can redirect the output to a file (for example, `show tech-support diagnostic > flash:filename`) in the local writable storage file system or remote file system.

```
Note
```

For devices that support stacking, this command is executed on every switch that is up. For devices that do not support stacking, this command is executed only on the active switch.

The output of this command displays the output of the following commands:

- `show clock`
- `show version`
- `show running-config`
- `show inventory`
- `show diagnostic bootup level`
- `show diagnostic status`
- `show diagnostic content switch all`
- `show diagnostic result switch all detail`
- `show diagnostic schedule switch all`
- `show diagnostic post`
- `show diagnostic description switch [switch number] test all`
- `show logging onboard switch [switch number] cli log detail`
- `show logging onboard switch [switch number] counter detail`
- `show logging onboard switch [switch number] environment detail`
- `show logging onboard switch [switch number] message detail`
• show logging onboard switch [switch number] poe detail
• show logging onboard switch [switch number] status
• show logging onboard switch [switch number] temperature detail
• show logging onboard switch [switch number] uptime detail
• show logging onboard switch [switch number] voltage detail

Examples

The following is a sample output from the show tech-support diagnostic command:

Device# show tech-support diagnostic
.
.
.
------------------ show diagnostic status ------------------

<BU> - Bootup Diagnostics, <HM> - Health Monitoring Diagnostics,
<OD> - OnDemand Diagnostics, <SCH> - Scheduled Diagnostics

<table>
<thead>
<tr>
<th>Card</th>
<th>Description</th>
<th>Current Running Test</th>
<th>Run by</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>C3650</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>2</td>
<td>MODEL UNSET</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>3</td>
<td>MODEL UNSET</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>4</td>
<td>MODEL UNSET</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>5</td>
<td>MODEL UNSET</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>6</td>
<td>MODEL UNSET</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>7</td>
<td>MODEL UNSET</td>
<td>N/A</td>
<td>N/A</td>
</tr>
</tbody>
</table>

------------------ show diagnostic status ------------------
### switch 1:

Diagnostics test suite attributes:

- M/C/*: Minimal bootup level test / Complete bootup level test / NA
- B/*: Basic ondemand test / NA
- P/V/*: Per port test / Per device test / NA
- D/N/*: Disruptive test / Non-disruptive test / NA
- S/*: Only applicable to standby unit / NA
- X/*: Not a health monitoring test / NA
- F/*: Fixed monitoring interval test / NA
- E/*: Always enabled monitoring test / NA
- A/I: Monitoring is active / Monitoring is inactive

<table>
<thead>
<tr>
<th>ID</th>
<th>Test Name</th>
<th>Attributes</th>
<th>Test Interval</th>
<th>Test Throttle</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>DiagGoldPktTest</td>
<td><em>BPN</em>X**I</td>
<td>not configured</td>
<td>n/a</td>
</tr>
<tr>
<td>2</td>
<td>DiagThermalTest</td>
<td><em>B</em>N****A</td>
<td>000 00:01:30.00 5</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>DiagFanTest</td>
<td><em>B</em>N****A</td>
<td>000 00:01:30.00 5</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>DiagPhyLoopbackTest</td>
<td><em>BPD</em>X**I</td>
<td>not configured</td>
<td>n/a</td>
</tr>
<tr>
<td>5</td>
<td>DiagScratchRegisterTest</td>
<td><em>B</em>N****A</td>
<td>000 00:01:30.00 5</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>TestUnusedPortLoopback</td>
<td><em>BPN</em>***I</td>
<td>not configured</td>
<td>n/a</td>
</tr>
<tr>
<td>7</td>
<td>TestPortTxMonitoring</td>
<td><em>BPN</em>***A</td>
<td>000 00:01:30.00 1</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>DiagPoETest</td>
<td><strong><em>D</em>X</strong>I</td>
<td>not configured</td>
<td>n/a</td>
</tr>
<tr>
<td>9</td>
<td>DiagStackCableTest</td>
<td><strong><em>D</em>X</strong>I</td>
<td>not configured</td>
<td>n/a</td>
</tr>
<tr>
<td>10</td>
<td>DiagMemoryTest</td>
<td><em>B</em>D*X**I</td>
<td>not configured</td>
<td>n/a</td>
</tr>
</tbody>
</table>

### switch 2:

Diagnostics test suite attributes:
<table>
<thead>
<tr>
<th>ID</th>
<th>Test Name</th>
<th>Attributes</th>
<th>Test Interval</th>
<th>Thre-</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>DiagGoldPktTest</td>
<td><em>BPN</em>X**I</td>
<td>not configured</td>
<td>n/a</td>
</tr>
<tr>
<td>2</td>
<td>DiagThermalTest</td>
<td><em>B</em>N****A</td>
<td>000 00:01:30.00</td>
<td>5</td>
</tr>
<tr>
<td>3</td>
<td>DiagFanTest</td>
<td><em>B</em>N****A</td>
<td>000 00:01:30.00</td>
<td>5</td>
</tr>
<tr>
<td>4</td>
<td>DiagPhyLoopbackTest</td>
<td><em>BPD</em>X**I</td>
<td>not configured</td>
<td>n/a</td>
</tr>
<tr>
<td>5</td>
<td>DiagScratchRegisterTest</td>
<td><em>B</em>N****A</td>
<td>000 00:01:30.00</td>
<td>5</td>
</tr>
<tr>
<td>6</td>
<td>TestUnusedPortLoopback</td>
<td><em>BPN</em>***I</td>
<td>not configured</td>
<td>n/a</td>
</tr>
<tr>
<td>7</td>
<td>TestPortTxMonitoring</td>
<td><em>BPN</em>***A</td>
<td>000 00:01:30.00</td>
<td>1</td>
</tr>
<tr>
<td>8</td>
<td>DiagPoETest</td>
<td><strong><em>D</em>X</strong>I</td>
<td>not configured</td>
<td>n/a</td>
</tr>
<tr>
<td>9</td>
<td>DiagStackCableTest</td>
<td><strong><em>D</em>X</strong>I</td>
<td>not configured</td>
<td>n/a</td>
</tr>
<tr>
<td>10</td>
<td>DiagMemoryTest</td>
<td><em>B</em>D*X**I</td>
<td>not configured</td>
<td>n/a</td>
</tr>
</tbody>
</table>

show logging onboard switch 4 cli log detail

---

CLI LOGGING SUMMARY INFORMATION

COUNT COMMAND

No summary data to display
Interface and Hardware Commands

show tech-support diagnostic

---------------------------------------------------------------------
CLI LOGGING CONTINUOUS INFORMATION
---------------------------------------------------------------------
MM/DD/YYYY HH:MM:SS COMMAND
---------------------------------------------------------------------
No continuous data
---------------------------------------------------------------------

------------------ show logging onboard switch 5 clilog detail --------------

---------------------------------------------------------------------
CLI LOGGING SUMMARY INFORMATION
---------------------------------------------------------------------
COUNT COMMAND
---------------------------------------------------------------------
No summary data to display
---------------------------------------------------------------------

------------------ show tech-support diagnostic --------------

---------------------------------------------------------------------
CLI LOGGING CONTINUOUS INFORMATION
---------------------------------------------------------------------
MM/DD/YYYY HH:MM:SS COMMAND
---------------------------------------------------------------------
No continuous data
---------------------------------------------------------------------

. 
. 
.
speed

To specify the speed of a 10/100/1000/2500/5000 Mbps port, use the `speed` command in interface configuration mode. To return to the default value, use the `no` form of this command.

```
speed {10 | 100 | 1000 | 2500 | 5000 | auto [10 | 100 | 1000 | 2500 | 5000]} | nonegotiate
```

<table>
<thead>
<tr>
<th>Syntax Description</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>Specifies that the port runs at 10 Mbps.</td>
</tr>
<tr>
<td>100</td>
<td>Specifies that the port runs at 100 Mbps.</td>
</tr>
<tr>
<td>1000</td>
<td>Specifies that the port runs at 1000 Mbps. This option is valid and visible only on 10/100/1000 Mb/s ports.</td>
</tr>
<tr>
<td>2500</td>
<td>Specifies that the port runs at 2500 Mbps. This option is valid and visible only on multi-Gigabit-supported Ethernet ports.</td>
</tr>
<tr>
<td>5000</td>
<td>Specifies that the port runs at 5000 Mbps. This option is valid and visible only on multi-Gigabit-supported Ethernet ports.</td>
</tr>
<tr>
<td>auto</td>
<td>Detects the speed at which the port should run, automatically, based on the port at the other end of the link. If you use the 10, 100, 1000, 2500, or 5000 keyword with the auto keyword, the port autonegotiates only at the specified speeds.</td>
</tr>
<tr>
<td>nonegotiate</td>
<td>Disables autonegotiation, and the port runs at 1000 Mbps.</td>
</tr>
</tbody>
</table>

Command Default

The default is `auto`.

Command Modes

Interface configuration

Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE 3.2SE</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>Cisco IOS XE Denali 16.3.1</td>
<td>This command was modified. The following keywords were added: 2500 and 5000. These keywords are visible only on multi-Gigabit Ethernet port supporting devices.</td>
</tr>
</tbody>
</table>

Usage Guidelines

You cannot configure speed on 10-Gigabit Ethernet ports.

Except for the 1000BASE-T small form-factor pluggable (SFP) modules, you can configure the speed to not negotiate (nonegotiate) when an SFP module port is connected to a device that does not support autonegotiation.

The new keywords, 2500 and 5000 are visible only on multi-Gigabit (m-Gig) Ethernet supporting devices.

If the speed is set to auto, the switch negotiates with the device at the other end of the link for the speed setting, and then forces the speed setting to the negotiated value. The duplex setting remains configured on each end of the link, which might result in a duplex setting mismatch.
If both ends of the line support autonegotiation, we highly recommend the default autonegotiation settings. If one interface supports autonegotiation and the other end does not, use the auto setting on the supported side, but set the duplex and speed on the other side.

⚠️ Caution

Changing the interface speed and duplex mode configuration might shut down and re-enable the interface during the reconfiguration.

For guidelines on setting the switch speed and duplex parameters, see the “Configuring Interface Characteristics” chapter in the software configuration guide for this release.

Verify your settings using the `show interfaces` privileged EXEC command.

**Examples**

The following example shows how to set speed on a port to 100 Mbps:

```
Device(config)# interface gigabitethernet1/0/1
Device(config-if)# speed 100
```

The following example shows how to set a port to autonegotiate at only 10 Mbps:

```
Device(config)# interface gigabitethernet1/0/1
Device(config-if)# speed auto 10
```

The following example shows how to set a port to autonegotiate at only 10 or 100 Mbps:

```
Device(config)# interface gigabitethernet1/0/1
Device(config-if)# speed auto 10 100
```
To configure StackPower parameters for the power stack or for a switch in the power stack, use the `stack power` command in global configuration mode. To return to the default setting, use the `no` form of the command,

```
stack-power {stack power-stack-name | switch stack-member-number}
no stack-power {stack power-stack-name | switch stack-member-number}
```

**Syntax Description**

- `stack power-stack-name` Specifies the name of the power stack. The name can be up to 31 characters. Entering these keywords followed by a carriage return enters power stack configuration mode.

- `switch stack-member-number` Specifies the switch number in the stack (1 to 4) to enter switch stack-power configuration mode for the switch.

**Command Default**

There is no default.

**Command Modes**

Global configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE 3.2SE</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

When you enter the `stack-power stack power stack name` command, you enter power stack configuration mode, and these commands are available:

- `default`—Returns a command to its default setting.
- `exit`—Exits ARP access-list configuration mode.
- `mode`—Sets the power mode for the power stack. See the `mode` command.
- `no`—Negates a command or returns to default settings.

If you enter the `stack-power switch switch-number` command with a switch number that is not participating in StackPower, you receive an error message.

When you enter the `stack-power switch switch-number` command with the number of a switch participating in StackPower, you enter switch stack-power configuration mode, and these commands are available:

- `default`—Returns a command to its default setting.
- `exit`—Exits switch stack power configuration mode.
- `no`—Negates a command or returns to default settings.
- `power-priority`—Sets the power priority for the switch and the switch ports. See the `power-priority` command.
- `stack-id name`—Enters the name of the power stack to which the switch belongs. If you do not enter the power stack-ID, the switch does not inherit the stack parameters. The name can be up to 31 characters.
- `standalone`—Forces the switch to operate in standalone power mode. This mode shuts down both stack power ports.

**Examples**

This example removes switch 2, which is connected to the power stack, from the power pool and shutting down both power ports:

```
```
Device(config)# stack-power switch 2
Device(config-switch-stackpower)# standalone
Device(config-switch-stackpower)# exit
**switchport block**

To prevent unknown multicast or unicast packets from being forwarded, use the `switchport block` command in interface configuration mode. To allow forwarding unknown multicast or unicast packets, use the `no` form of this command.

```
switchport block {multicast | unicast}
no switchport block {multicast | unicast}
```

### Syntax Description

- **multicast** Specifies that unknown multicast traffic should be blocked.

  **Note** Only pure Layer 2 multicast traffic is blocked. Multicast packets that contain IPv4 or IPv6 information in the header are not blocked.

- **unicast** Specifies that unknown unicast traffic should be blocked.

### Command Default

Unknown multicast and unicast traffic is not blocked.

### Command Modes

Interface configuration

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE 3.2SE</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

By default, all traffic with unknown MAC addresses is sent to all ports. You can block unknown multicast or unicast traffic on protected or nonprotected ports. If unknown multicast or unicast traffic is not blocked on a protected port, there could be security issues.

With multicast traffic, the port blocking feature blocks only pure Layer 2 packets. Multicast packets that contain IPv4 or IPv6 information in the header are not blocked.

Blocking unknown multicast or unicast traffic is not automatically enabled on protected ports; you must explicitly configure it.

For more information about blocking packets, see the software configuration guide for this release.

This example shows how to block unknown unicast traffic on an interface:

```
Device(config-if)# switchport block unicast
```

You can verify your setting by entering the `show interfaces interface-id switchport` privileged EXEC command.
system mtu

**Syntax Description**

`bytes`

**Command Default**

The default MTU size for all ports is 1500 bytes.

**Command Modes**

Global configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE 3.2SE</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

You can verify your setting by entering the `show system mtu` privileged EXEC command.

The switch does not support the MTU on a per-interface basis.

If you enter a value that is outside the allowed range for the specific type of interface, the value is not accepted.
test mcu read-register

To enable debugging of the Power over Ethernet (PoE) controller, use the test mcu read-register command in privileged EXEC mode.

test mcu read-register  {det-cls-offset | manufacture-id | port-mode}

**Syntax Description**
- **det-cls-offset**: Displays the read detection classification register summary.
- **manufacture-id**: Displays the PoE controller manufacture ID.
- **port-mode**: Displays the port mode details.

**Command Modes**
Privileged EXEC

**Command History**
<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE 3.2SE</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Examples**
The following is sample output from the test mcu read-register  det-cls-offset command:

```
Device# test mcu read-register det-cls-offset 1
DETECTION ENABLE BIT SUMMARY

<table>
<thead>
<tr>
<th>Controller</th>
<th>port1</th>
<th>port2</th>
<th>port3</th>
<th>port4</th>
<th>register (hexadecimal)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>5</td>
</tr>
</tbody>
</table>

CLASSIFICATION ENABLE BIT SUMMARY

<table>
<thead>
<tr>
<th>Controller</th>
<th>port1</th>
<th>port2</th>
<th>port3</th>
<th>port4</th>
<th>register (hexadecimal)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>
```
The following is sample output from the `test mcu read-register` `manufacture-id` command:

```
MANUFACTURE ID : DEVICE_BCM_PALPATINE reg_val = 0x1B
```

The following is sample output from the `test mcu read-register` `port-mode` command:

```
PORT MODE SUMMERY
Controller | port1 | port2 | port3 | port4 | register (hexadecimal)
-----------|-------|-------|-------|-------|----------------------
1          | 01    | 00    | 01    | 00    | 22                   
2          | 01    | 00    | 01    | 00    | 22                   
3          | 01    | 00    | 01    | 00    | 22                   
4          | 01    | 00    | 01    | 00    | 22                   
5          | 01    | 00    | 01    | 00    | 22                   
6          | 01    | 00    | 01    | 00    | 22                   
7          | 01    | 00    | 01    | 00    | 22                   
8          | 01    | 00    | 01    | 00    | 22                   
9          | 01    | 00    | 01    | 00    | 22                   
10         | 01    | 00    | 01    | 00    | 22                   
11         | 00    | 00    | 01    | 00    | 20                   
12         | 01    | 00    | 00    | 00    | 2                    
```
voice-signaling vlan (network-policy configuration)

To create a network-policy profile for the voice-signaling application type, use the `voice-signaling vlan` command in network-policy configuration mode. To delete the policy, use the `no` form of this command.

```
voice-signaling vlan {vlan-id [ {cos cos-value | dscp dscp-value} ] | dot1p [ {cos l2-priority | dscp dscp} ] | none | untagged}
```

**Syntax Description**

- **vlan-id** (Optional) The VLAN for voice traffic. The range is 1 to 4094.
- **cos cos-value** (Optional) Specifies the Layer 2 priority class of service (CoS) for the configured VLAN. The range is 0 to 7; the default is 5.
- **dscp dscp-value** (Optional) Specifies the differentiated services code point (DSCP) value for the configured VLAN. The range is 0 to 63; the default is 46.
- **dot1p** (Optional) Configures the phone to use IEEE 802.1p priority tagging and to use VLAN 0 (the native VLAN).
- **none** (Optional) Does not instruct the Cisco IP phone about the voice VLAN. The phone uses the configuration from the phone key pad.
- **untagged** (Optional) Configures the phone to send untagged voice traffic. This is the default for the phone.

**Command Default**

No network-policy profiles for the voice-signaling application type are defined.

The default CoS value is 5.

The default DSCP value is 46.

The default tagging mode is untagged.

**Command Modes**

Network-policy profile configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE 3.2SE</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `network-policy profile` global configuration command to create a profile and to enter network-policy profile configuration mode.

The voice-signaling application type is for network topologies that require a different policy for voice signaling than for voice media. This application type should not be advertised if all of the same network policies apply as those advertised in the voice policy TLV.

When you are in network-policy profile configuration mode, you can create the profile for voice-signaling by specifying the values for VLAN, class of service (CoS), differentiated services code point (DSCP), and tagging mode.

These profile attributes are contained in the Link Layer Discovery Protocol for Media Endpoint Devices (LLDP-MED) network-policy time-length-value (TLV).
To return to privileged EXEC mode from the network-policy profile configuration mode, enter the `exit` command.

This example shows how to configure voice-signaling for VLAN 200 with a priority 2 CoS:
```
Device(config)# network-policy profile 1
Device(config-network-policy)# voice-signaling vlan 200 cos 2
```

This example shows how to configure voice-signaling for VLAN 400 with a DSCP value of 45:
```
Device(config)# network-policy profile 1
Device(config-network-policy)# voice-signaling vlan 400 dscp 45
```

This example shows how to configure voice-signaling for the native VLAN with priority tagging:
```
Device(config-network-policy)# voice-signaling vlan dot1p cos 4
```
To create a network-policy profile for the voice application type, use the **voice vlan** command in network-policy configuration mode. To delete the policy, use the **no** form of this command.

```
voice vlan {vlan-id [ {cos cos-value | dscp dscp-value} ] | dot1p [ {cos 12-priority | dscp dscp} ] | none | untagged}
```

**Syntax Description**

- **vlan-id** (Optional) The VLAN for voice traffic. The range is 1 to 4094.
- **cos cos-value** (Optional) Specifies the Layer 2 priority class of service (CoS) for the configured VLAN. The range is 0 to 7; the default is 5.
- **dscp dscp-value** (Optional) Specifies the differentiated services code point (DSCP) value for the configured VLAN. The range is 0 to 63; the default is 46.
- **dot1p** (Optional) Configures the phone to use IEEE 802.1p priority tagging and to use VLAN 0 (the native VLAN).
- **none** (Optional) Does not instruct the Cisco IP phone about the voice VLAN. The phone uses the configuration from the phone key pad.
- **untagged** (Optional) Configures the phone to send untagged voice traffic. This is the default for the phone.

**Command Default**

No network-policy profiles for the voice application type are defined.

- The default CoS value is 5.
- The default DSCP value is 46.
- The default tagging mode is untagged.

**Command Modes**

Network-policy profile configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE 3.2SE</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the **network-policy profile** global configuration command to create a profile and to enter network-policy profile configuration mode.

The voice application type is for dedicated IP telephones and similar devices that support interactive voice services. These devices are typically deployed on a separate VLAN for ease of deployment and enhanced security through isolation from data applications.

When you are in network-policy profile configuration mode, you can create the profile for voice by specifying the values for VLAN, class of service (CoS), differentiated services code point (DSCP), and tagging mode.

These profile attributes are contained in the Link Layer Discovery Protocol for Media Endpoint Devices (LLDP-MED) network-policy time-length-value (TLV).
To return to privileged EXEC mode from the network-policy profile configuration mode, enter the `exit` command.

This example shows how to configure the voice application type for VLAN 100 with a priority 4 CoS:

```
Device(config)# network-policy profile 1
Device(config-network-policy)# voice vlan 100 cos 4
```

This example shows how to configure the voice application type for VLAN 100 with a DSCP value of 34:

```
Device(config)# network-policy profile 1
Device(config-network-policy)# voice vlan 100 dscp 34
```

This example shows how to configure the voice application type for the native VLAN with priority tagging:

```
Device(config-network-policy)# voice vlan dot1p cos 4
```
voice vlan (network-policy configuration)