# Onboard Failure Logging

**First Published: August 10, 2007**
**Last Updated: August 10, 2007**

The Onboard Failure Logging (OBFL) feature collects data such as operating temperatures, hardware uptime, interrupts, and other important events and messages from system hardware installed in a Cisco router or switch. The data is stored in nonvolatile memory and helps technical personnel diagnose hardware problems.

### Finding Feature Information in This Module

Your Cisco IOS software release may not support all of the features documented in this module. To reach links to specific feature documentation in this module and to see a list of the releases in which each feature is supported, use the "Feature Information for OBFL" section on page 27.

### Finding Support Information for Platforms and Cisco IOS and Catalyst OS Software Images

Use Cisco Feature Navigator to find information about platform support and Cisco IOS and Catalyst OS software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

## Contents



**Americas Headquarters:**
**Cisco Systems, Inc., 170 West Tasman Drive, San Jose, CA 95134-1706 USA**

# Restrictions for OBFL

### Software Restrictions

If a device (router or switch) intends to use *linear* flash memory as its OBFL storage media, Cisco IOS software must reserve a minimum of two physical sectors (or physical blocks) for the OBFL feature. Because an erase operation for a linear flash device is done on per-sector (or per-block) basis, one extra physical sector is needed. Otherwise, the minimum amount of space reserved for the OBFL feature on any device must be at least 8 KB.

### Firmware Restrictions

If a line card or port adapter runs an operating system or firmware that is different from the Cisco IOS operating system, the line card or port adapter must provide device driver level support or an interprocess communications (IPC) layer that allows the OBFL file system to communicate to the line card or port adapter. This requirement is enforced to allow OBFL data to be recorded on a storage device attached to the line card or port adapter.

### Hardware Restrictions

To support the OBFL feature, a device must have at least 8 KB of nonvolatile memory space reserved for OBFL data logging.

# Information About OBFL

To use the OBFL feature, you should understand the following concept:

- Data Collected

# Data Collected

The OBFL feature records operating temperatures, hardware uptime, interrupts, and other important events and messages that can assist with diagnosing problems with hardware cards (or *modules*) installed in a Cisco router or switch. Data is logged to files stored in nonvolatile memory. When the onboard hardware is started up, a first record is made for each area monitored and becomes a base value for subsequent records. The OBFL feature provides a circular updating scheme for collecting continuous records and archiving older (historical) records, ensuring accurate data about the system. Data is recorded in one of two formats: continuous information that displays a snapshot of measurements and samples in a continuous file, and summary information that provides details about the data being collected. The data is displayed using the **show logging onboard** command. The message “No historical data to display” is seen when historical data is not available.

The following sections describe the type of data collected in more detail.

## Temperature

Temperatures surrounding hardware modules can exceed recommended safe operating ranges and cause system problems such as packet drops. Higher than recommended operating temperatures can also accelerate component degradation and affect device reliability. Monitoring temperatures is important for maintaining environmental control and system reliability. Once a temperature sample is logged, the

sample becomes the base value for the next record. From that point on, temperatures are recorded either when there are changes from the previous record or if the maximum storage time is exceeded. Temperatures are measured and recorded in degrees Celsius.

### Temperature Example

```
--------------------------------------------------------------------------------
TEMPERATURE SUMMARY INFORMATION
--------------------------------------------------------------------------------
Number of sensors          : 12
Sampling frequency         : 5 minutes
Maximum time of storage    : 120 minutes
--------------------------------------------------------------------------------
Sensor                             |   ID  | Maximum Temperature 0C
--------------------------------------------------------------------------------
MB-Out                               980201     43
MB-In                                980202     28
MB                                   980203     29
MB                                   980204     38
EARL-Out                             910201     0
EARL-In                              910202     0
SSA 1                                980301     38
SSA 2                                980302     36
JANUS 1                              980303     36
JANUS 2                              980304     35
GEMINI 1                             980305     0
GEMINI 2                             980306     0
-----------------------------------------------------------------
Temp                          Sensor ID
0C     1    2    3    4    5    6    7    8    9   10   11   12
-----------------------------------------------------------------
No historical data to display
-----------------------------------------------------------------
--------------------------------------------------------------------------------
TEMPERATURE CONTINUOUS INFORMATION
--------------------------------------------------------------------------------
Sensor                             |   ID  |
--------------------------------------------------------------------------------
MB-Out                               980201
MB-In                                980202
MB                                   980203
MB                                   980204
EARL-Out                             910201
EARL-In                              910202
SSA 1                                980301
SSA 2                                980302
JANUS 1                              980303
JANUS 2                              980304
GEMINI 1                             980305
GEMINI 2                             980306

-------------------------------------------------------------------------------
       Time Stamp   |Sensor Temperature 0C
MM/DD/YYYY HH:MM:SS |  1    2    3    4    5    6    7    8    9   10   11   12
-------------------------------------------------------------------------------
03/06/2007 22:32:51   31   26   27   27   NA   NA   33   32   30   29   NA   NA
03/06/2007 22:37:51   43   28   29   38   NA   NA   38   36   36   35   NA   NA
-------------------------------------------------------------------------------
```

To interpret this data:

- Number of sensors is the total number of temperature sensors that will be recorded. A column for each sensor is displayed with temperatures listed under the number of each sensor, as available.
- Sampling frequency is the time between measurements.
- Maximum time of storage determines the maximum amount of time, in minutes, that can pass when the temperature remains unchanged and the data is not saved to storage media. After this time, a temperature record will be saved even if the temperature has not changed.
- The Sensor column lists the name of the sensor.
- The ID column lists an assigned identifier for the sensor.
- Maximum Temperature 0C shows the highest recorded temperature per sensor.
- Temp indicates a recorded temperature in degrees Celsius in the historical record. Columns following show the total time each sensor has recorded that temperature.
- Sensor ID is an assigned number, so that temperatures for the same sensor can be stored together.

## Operational Uptime

The operational uptime tracking begins when the module is powered on, and information is retained for the life of the module.

### Operational Uptime Example

```
--------------------------------------------------------------------------------
UPTIME SUMMARY INFORMATION
--------------------------------------------------------------------------------
First customer power on : 03/06/2007 22:32:51
Total uptime            :   0 years   0 weeks   2 days  18 hours  10 minutes
Total downtime          :   0 years   0 weeks   0 days   8 hours   7 minutes
Number of resets        : 130
Number of slot changes  : 16
Current reset reason    : 0xA1
Current reset timestamp : 03/07/2007 13:29:07
Current slot            : 2
Current uptime          :   0 years   0 weeks   1 days   7 hours   0 minutes
--------------------------------------------------------------------------------
Reset  |        |
Reason | Count  |
--------------------------------------------------------------------------------
0x5         64
0x6         62
0xA1         4
--------------------------------------------------------------------------------
--------------------------------------------------------------------------------
UPTIME CONTINUOUS INFORMATION
--------------------------------------------------------------------------------
Time Stamp          | Reset  | Uptime
MM/DD/YYYY HH:MM:SS | Reason | years weeks days hours  minutes
--------------------------------------------------------------------------------
03/06/2007 22:32:51   0xA1      0     0     0     0     0
--------------------------------------------------------------------------------
```

The operational uptime application tracks the following events:

- Date and time the customer first powered on a component.
- Total uptime and downtime for the component in years, weeks, days, hours, and minutes.

- Total number of component resets.
- Total number of slot (module) changes.
- Current reset timestamp to include the date and time.
- Current slot (module) number of the component.
- Current uptime in years, weeks, days, hours, and minutes.
- Reset reason; see Table 1 to translate the numbers displayed.
- Count is the number of resets that have occurred for each reset reason.

*Table 1 Reset Reason Codes and Explanations*

| Reset Reason Code (in hex) | Component/Explanation |
|---|---|
| 0x01 | Chassis on |
| 0x02 | Line card hot plug in |
| 0x03 | Supervisor requests line card off or on |
| 0x04 | Supervisor requests hard reset on line card |
| 0x05 | Line card requests Supervisor off or on |
| 0x06 | Line card requests hard reset on Supervisor |
| 0x07 | Line card self reset using the internal system register |
| 0x08 | — |
| 0x09 | — |
| 0x0A | Momentary power interruption on the line card |
| 0x0B | — |
| 0x0C | — |
| 0x0D | — |
| 0x0E | — |
| 0x0F | — |
| 0x10 | — |
| 0x11 | Off or on after Supervisor non-maskable interrupts (NMI) |
| 0x12 | Hard reset after Supervisor NMI |
| 0x13 | Soft reset after Supervisor NMI |
| 0x14 | — |
| 0x15 | Off or on after line card asks Supervisor NMI |
| 0x16 | Hard reset after line card asks Supervisor NMI |
| 0x17 | Soft reset after line card asks Supervisor NMI |
| 0x18 | — |
| 0x19 | Off or on after line card self NMI |
| 0x1A | Hard reset after line card self NMI |
| 0x1B | Soft reset after line card self NMI |

*Table 1 Reset Reason Codes and Explanations*

| Reset Reason Code (in hex) | Component/Explanation |
|---|---|
| 0x21 | Off or on after spurious NMI |
| 0x22 | Hard reset after spurious NMI |
| 0x23 | Soft reset after spurious NMI |
| 0x24 | — |
| 0x25 | Off or on after watchdog NMI |
| 0x26 | Hard reset after watchdog NMI |
| 0x27 | Soft reset after watchdog NMI |
| 0x28 | — |
| 0x29 | Off or on after parity NMI |
| 0x2A | Hard reset after parity NMI |
| 0x2B | Soft reset after parity NMI |
| 0x31 | Off or on after system fatal interrupt |
| 0x32 | Hard reset after system fatal interrupt |
| 0x33 | Soft reset after system fatal interrupt |
| 0x34 | — |
| 0x35 | Off or on after application-specific integrated circuit (ASIC) interrupt |
| 0x36 | Hard reset after ASIC interrupt |
| 0x37 | Soft reset after ASIC interrupt |
| 0x38 | — |
| 0x39 | Off or on after unknown interrupt |
| 0x3A | Hard reset after unknown interrupt |
| 0x3B | Soft reset after unknown interrupt |
| 0x41 | Off or on after CPU exception |
| 0x42 | Hard reset after CPU exception |
| 0x43 | Soft reset after CPU exception |
| 0xA1 | Reset data converted to generic data |

## Interrupts

Interrupts are generated by system components that require attention from the CPU such as ASICs and NMIs. Interrupts are generally related to hardware limit conditions or errors that need to be corrected.

The continuous format records each time a component is interrupted, and this record is stored and used as base information for subsequent records. Each time the list is saved, a timestamp is added. Time differences from the previous interrupt are counted, so that technical personnel can gain a complete record of the component's operational history when an error occurs.

#### Interrupts Example

```
------------------------------------------------------------------------------
INTERRUPT SUMMARY INFORMATION
------------------------------------------------------------------------------
Name                                              |  ID | Offset | Bit |  Count
------------------------------------------------------------------------------
No historical data to display
------------------------------------------------------------------------------
------------------------------------------------------------------------------
CONTINUOUS INTERRUPT INFORMATION
------------------------------------------------------------------------------
MM/DD/YYYY HH:MM:SS mmm | Name                                |  ID | Offset | Bit
------------------------------------------------------------------------------
03/06/2007 22:33:06 450   Port-ASIC #2                          9   0x00E7     6
------------------------------------------------------------------------------
```

To interpret this data:

- Name is a description of the component including its position in the device.
- ID is an assigned field for data storage.
- Offset is the register offset from a component register's base address.
- Bit is the interrupt bit number recorded from the component's internal register.
- The timestamp shows the date and time that an interrupt occurred down to the millisecond.

## Message Logging

The OBFL feature logs standard system messages. Instead of displaying the message to a terminal, the message is written to and stored in a file, so the message can be accessed and read at a later time. System messages range from level 1 alerts to level 7 debug messages, and these levels can be specified in the **hw module logging onboard** command.

#### Error Message Log Example

```
------------------------------------------------------------------------------
ERROR MESSAGE SUMMARY INFORMATION
------------------------------------------------------------------------------
Facility-Sev-Name      | Count | Persistence Flag
MM/DD/YYYY HH:MM:SS
------------------------------------------------------------------------------
No historical data to display
------------------------------------------------------------------------------
------------------------------------------------------------------------------
ERROR MESSAGE CONTINUOUS INFORMATION
------------------------------------------------------------------------------
MM/DD/YYYY HH:MM:SS Facility-Sev-Name
------------------------------------------------------------------------------
03/06/2007 22:33:35  %GOLD_OBFL-3-GOLD : Diagnostic OBFL: Diagnostic OBFL testing
```

To interpret this data:

- A timestamp shows the date and time the message was logged.
- Facility-Sev-Name is a coded naming scheme for a system message, as follows:
  - The Facility code consists of two or more uppercase letters that indicate the hardware device (facility) to which the message refers.
  - Sev is a single-digit code from 1 to 7 that reflects the severity of the message.

  - Name is one or two code names separated by a hyphen that describe the part of the system from where the message is coming.
- The error message follows the Facility-Sev-Name codes. For more information about system messages, see the *Cisco IOS System and Error Messages* guide.
- Count indicates the number of instances of this message that is allowed in the history file. Once that number of instances has been recorded, the oldest instance will be removed from the history file to make room for new ones.
- The Persistence Flag gives a message priority over others that do not have the flag set.

# How to Enable OBFL

This section contains the following procedure:

- Enabling OBFL

# Enabling OBFL

The OBFL feature is enabled by default. Because of the valuable information this feature offers technical personnel, it should not be disabled. If you find the feature has been disabled, use the following steps to reenable it.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **hw-module switch** *switch-number* **module** *module-number* **logging onboard** [**message level** {**1**-**7**}]
4. **end**

### DETAILED STEPS

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | `enable`<br><br>**Example:**<br>`Router> enable` | Enables privileged EXEC mode.<br>• Enter your password if prompted. |
| **Step 2** | `configure terminal`<br><br>**Example:**<br>`Router# configure terminal` | Enters global configuration mode. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 3** | **hw-module switch** *switch-number* **module** *module-number* **logging onboard** [**message level** {**1**-**7**}]<br><br>**Example:**<br>`Router(config)# hw-module switch 2 module 1 logging onboard` | Enables OBFL on the specified hardware module.<br><br>**Note** By default, all system messages sent to a device are logged by the OBFL feature. You can define a specific message level (only level 1 messages, as an example) to be logged using the **message level** keywords. |
| **Step 4** | **end**<br><br>**Example:**<br>`Router(config)# end` | Ends global configuration mode. |

# Configuration Examples for OBFL

The important OBFL feature is the information that is displayed by the **show logging onboard module** privileged EXEC command. This section provides the following examples of how to enable and display OBFL records.

- Enabling OBFL Message Logging: Example
- OBFL Message Log: Example
- OBFL Component Uptime Report: Example
- OBFL Report for a Specific Time: Example

## Enabling OBFL Message Logging: Example

The following example shows how to configure OBFL message logging at level 3:

```
hw-module switch 2 module 1 logging onboard message level 3
```

## OBFL Message Log: Example

The following example shows how to display the system messages that are being logged for module 2:

```
Router# show logging onboard module 2 message continuous

--------------------------------------------------------------------------------
ERROR MESSAGE CONTINUOUS INFORMATION
--------------------------------------------------------------------------------
MM/DD/YYYY HH:MM:SS Facility-Sev-Name
--------------------------------------------------------------------------------
03/06/2007 22:33:35 %SWITCH_IF-3-CAMERR : [chars], for VCI [dec] VPI [dec] in stdby data
path check, status: [dec]
--------------------------------------------------------------------------------
```

# OBFL Component Uptime Report: Example

The following example shows how to display a summary report for component uptimes for module 2:

```
Router# show logging onboard module 2 uptime

--------------------------------------------------------------------------------
UPTIME SUMMARY INFORMATION
--------------------------------------------------------------------------------
First customer power on : 03/06/2007 22:32:51
Total uptime            :   0 years   0 weeks   0 days   0 hours  35 minutes
Total downtime          :   0 years   0 weeks   0 days   0 hours   0 minutes
Number of resets        : 1
Number of slot changes  : 0
Current reset reason    : 0xA1
Current reset timestamp : 03/06/2007 22:31:34
Current slot            : 2
Current uptime          :   0 years   0 weeks   0 days   0 hours  35 minutes
--------------------------------------------------------------------------------
Reset  |        |
Reason | Count  |
--------------------------------------------------------------------------------
No historical data to display
--------------------------------------------------------------------------------
```

# OBFL Report for a Specific Time: Example

The following example shows how to display continuous reports for all components during a specific time period:

```
Router# show logging onboard module 3 continuous start 15:01:57 1 Mar 2007 end 15:04:57 3
Mar 2007

PID: WS-X6748-GE-TX    , VID:    , SN: SAL09063B85

--------------------------------------------------------------------------------
UPTIME CONTINUOUS INFORMATION
--------------------------------------------------------------------------------
Time Stamp          | Reset  | Uptime
MM/DD/YYYY HH:MM:SS | Reason | years weeks days hours  minutes
--------------------------------------------------------------------------------
03/01/2007 15:01:57   0xA1      0     0     0    10     0
03/03/2007 02:29:29   0xA1      0     0     0     5     0
--------------------------------------------------------------------------------

--------------------------------------------------------------------------------
TEMPERATURE CONTINUOUS INFORMATION
--------------------------------------------------------------------------------
Sensor                            |   ID  |
--------------------------------------------------------------------------------
MB-Out                              930201
MB-In                               930202
MB                                  930203
MB                                  930204
EARL-Out                            910201
EARL-In                             910202
SSA 1                               930301
SSA 2                               930302
JANUS 1                             930303
JANUS 2                             930304
GEMINI 1                            930305
```

```
GEMINI 2                              930306
--------------------------------------------------------------------------------
       Time Stamp    |Sensor Temperature 0C
MM/DD/YYYY HH:MM:SS |  1    2    3    4    5    6    7    8    9   10   11   12
--------------------------------------------------------------------------------
03/01/2007 15:01:57   26   26   NA   NA   NA   NA    0    0    0    0    0    0
03/01/2007 15:06:57   39   27   NA   NA   NA   NA   39   37   36   29   32   32
03/01/2007 15:11:02   40   27   NA   NA   NA   NA   40   38   37   30   32   32
03/01/2007 17:06:06   40   27   NA   NA   NA   NA   40   38   37   30   32   32
03/01/2007 19:01:09   40   27   NA   NA   NA   NA   40   38   37   30   32   32
03/03/2007 02:29:30   25   26   NA   NA   NA   NA    0    0    0    0    0    0
03/03/2007 02:34:30   38   26   NA   NA   NA   NA   39   37   36   29   31   31
03/03/2007 04:29:33   40   27   NA   NA   NA   NA   40   38   36   30   32   32
03/03/2007 06:24:37   40   27   NA   NA   NA   NA   40   38   36   29   32   32
03/03/2007 08:19:40   40   27   NA   NA   NA   NA   40   38   36   29   32   32
03/03/2007 10:14:44   40   27   NA   NA   NA   NA   40   38   36   30   32   32
03/03/2007 12:09:47   40   27   NA   NA   NA   NA   40   38   36   30   32   32
03/03/2007 14:04:51   40   27   NA   NA   NA   NA   40   38   36   30   32   32
--------------------------------------------------------------------------------
--------------------------------------------------------------------------------
CONTINUOUS INTERRUPT INFORMATION
--------------------------------------------------------------------------------
MM/DD/YYYY HH:MM:SS mmm | Name                              |  ID | Offset | Bit
--------------------------------------------------------------------------------
03/01/2007 15:01:59 350   Port-ASIC #0                        7   0x00E7     6
03/03/2007 02:29:34 650   Port-ASIC #0                        7   0x00E7     6
--------------------------------------------------------------------------------

--------------------------------------------------------------------------------
ERROR MESSAGE CONTINUOUS INFORMATION
--------------------------------------------------------------------------------
MM/DD/YYYY HH:MM:SS Facility-Sev-Name
--------------------------------------------------------------------------------
03/01/2007 15:02:15  %GOLD_OBFL-3-GOLD : Diagnostic OBFL: Diagnostic OBFL testing
03/03/2007 02:29:51  %GOLD_OBFL-3-GOLD : Diagnostic OBFL: Diagnostic OBFL testing
--------------------------------------------------------------------------------
```

# Additional References

The following sections provide references related to the OBFL feature.

## Related Documents

| Related Topic | Document Title |
|---|---|
| Onboard Failure Logging for Cisco 12000 series routers running Cisco IOS XR Software v3.4 | *Onboard Failure Logging on Cisco IOS Software Release 3.4* feature module |
| Onboard Failure Logging for Catalyst 3750-E and 3560-E switches running Cisco IOS Software Release 12.2(35)SE2 | "Using On-Board Failure Logging" section in the "Troubleshooting" chapter in the *Catalyst 3750-E and 3560-E Switch Software Configuration Guide*, 12.2(35)SE2 |
| Other related logging commands | *Catalyst 3750-E and 3560-E Switch Command Reference, Release 12.2(58)SE* |

## Standards

| Standard | Title |
|---|---|
| None | — |

## MIBs

| MIB | MIBs Link |
|---|---|
| None | To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL:<br>http://www.cisco.com/go/mibs |

## RFCs

| RFC | Title |
|---|---|
| None | — |

## Technical Assistance

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

# Command Reference

This section documents only commands that are new or modified:

- **clear logging onboard (Cat 6K)**
- **copy logging onboard (Cat 6K)**
- **hw-module logging onboard (Cat 6K)**
- **show logging onboard (Cat 6K)**

# clear logging onboard (Cat 6K)

To clear the onboard failure logs (OBFL) on Cisco Catalyst 6000 series switches, use the **clear logging onboard** command in privileged EXEC mode.

**clear logging onboard** [**module** *module-number*]

## Syntax Description

| | |
|---|---|
| **module** *module-number* | (Optional) Specifies a particular module. |

## Command Modes

Privileged EXEC (#)

## Command History

| Release | Modification |
|---|---|
| 12.2(33)SXH | This command was introduced. |

## Usage Guidelines

Use this command to clear all OBFL logs or only the logs in the module specified by the **module** *module-number* option.

**Note** Use this command with care: Important data could be lost when the logs are cleared. Make sure the logs have been transferred to a file before using this command.

## Examples

The following example shows how to clear the logs from module 2:

```
Router# clear logging onboard module 2
```

## Related Commands

| Command | Description |
|---|---|
| **attach** | Connects to a specific line card for the purpose of executing commands on that card. |
| **copy logging onboard module (Cat 6K)** | Copies OBFL data from the target OBFL-enabled module to a local or remote file system. |
| [**no**] **hw-module logging onboard (Cat 6K)** | Disables and enables OBFL. |
| **show logging onboard (Cat 6K)** | Displays onboard failure logs. |

# copy logging onboard (Cat 6K)

To copy onboard failure logging (OBFL) data from the target OBFL-enabled module in Cisco Catalyst 6000 series switches to a local or remote file system, use the **copy logging onboard** command in privileged EXEC mode.

**copy logging onboard module** *module-number destination-url*

## Syntax Description

| | |
|---|---|
| *module-number* | Specifies the module number. |
| *destination-url* | The destination URL of the copied file or directory. The destination can be either local or remote.<br>**Note** The exact format of the source and destination URLs varies according to the file or directory location. |

## Command Default

This command has no default condition.

## Command Modes

Privileged EXEC (#)

## Command History

| Release | Modification |
|---|---|
| 12.2(33)SXH | This command was introduced. |

## Usage Guidelines

This command copies OBFL data from the target OBFL-enabled board to a local or remote file system. See the *Cisco IOS Configuration Fundamentals Command Reference* for more information about use of the **copy** command.

## Examples

The following example shows the options for copying OBFL data:

```
Router# copy logging onboard module 2 ?

  bootflash:        Copy onboard logging to bootflash: file system
  const_nvram:      Copy onboard logging to const_nvram: file system
  dfc#2-bootflash:  Copy onboard logging to dfc#2-bootflash: file system
  dfc#4-bootflash:  Copy onboard logging to dfc#4-bootflash: file system
  disk0:            Copy onboard logging to disk0: file system
  disk1:            Copy onboard logging to disk1: file system
  ftp:              Copy onboard logging to ftp: file system
  http:             Copy onboard logging to http: file system
  https:            Copy onboard logging to https: file system
  null:             Copy onboard logging to null: file system
  nvram:            Copy onboard logging to nvram: file system
  rcp:              Copy onboard logging to rcp: file system
  scp:              Copy onboard logging to scp: file system
  sup-bootflash:    Copy onboard logging to sup-bootflash: file system
  sup-image:        Copy onboard logging to sup-image: file system
  syslog:           Copy onboard logging to syslog: file system
  system:           Copy onboard logging to system: file system
```

```
  tftp:             Copy onboard logging to tftp: file system
  tmpsys:           Copy onboard logging to tmpsys: file system
```

The following example shows how to transfer the OBFL data to a file on disk1:

```
Router# copy logging onboard module 2 disk1:tarmod2

OBFL feature copy disk1:tarmod2 2
% File transfer succeeded
```

The following example shows how to transfer the OBFL data to a file on a remote server:

```
Router# copy logging onboard module 2 tftp://server1/user1/tars/tarmod2/mod2tar

OBFL feature copy tftp://server1/user1/tars/tarmod2/mod2tar 2
% File transfer succeeded
```

## Related Commands

| Command | Description |
|---|---|
| **attach** | Connects to a specific line card for the purpose of executing commands on that card. |
| **clear logging onboard (Cat 6K)** | Clears onboard failure logs. |
| [**no**] **hw-module logging onboard (Cat 6K)** | Disables and enables OBFL. |
| **show logging onboard (Cat 6K)** | Displays onboard failure logs. |

# hw-module logging onboard (Cat 6K)

To re-enable onboard failure logging (OBFL) on Cisco Catalyst 6000 series switches if logging has been disabled, use the **hw-module logging onboard** command in global configuration mode. To disable OBFL (not recommended), use the **no** form of this command.

**hw-module switch** *switch-number* **module** *module-number* **logging onboard** [**message level** {**1**-**7**}]

**no hw-module switch** *switch-number* **module** *module-number* **logging onboard** [**message level** {**1**-**7**}]

## Syntax Description

| | |
|---|---|
| **switch** *switch-number* | Specifies the switch number. |
| **module** *module-number* | Specifies the module number. |
| **message level** {**1**-**7**} | (Optional) Specifies the level of severity for system messages that will be logged in OBFL files, as follows:<br>Level 1—Alert (immediate action needed)<br>Level 2—Critical condition<br>Level 3—Error condition<br>Level 4—Warning condition<br>Level 5—Notification (significant condition)<br>Level 6—Informational message only<br>Level 7—Debugging (appears during debugging only) |

## Command Default

Enabled in all hardware and is the recommended state; all levels of system messages are logged.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| 12.2(33)SXH | This command was introduced. |

## Usage Guidelines

This command enables operating temperatures, hardware uptime, interrupts, and other important events and messages to be recorded in files stored in nonvolatile memory, so that the data can be used to diagnose problems with hardware cards installed in a Cisco router or switch. When the onboard hardware is started up, a first record is made for each area monitored and becomes a base value for subsequent records. This command provides a circular updating scheme for collecting continuous records and archiving older (historical) records, ensuring accurate data about the hardware. Data is recorded in one of two formats: continuous information that displays a snapshot of data in a continuous file, and summary information that provides details about the data being collected. Use the **show logging onboard** privileged EXEC command to see reports of current and historical data.

This configuration command is applicable to the module inserted in a device. When the module is removed and inserted into a new device, the configuration of this command follows the module to the new device.

This command is normally accessed through the route processor or supervisor command line interface; however, some system images do not provide full support for client remote terminal access. When using these images, use the **attach** command to connect to the console on the line card.

## Examples

The following example shows how to configure OBFL message logging at level 7 (debugging):

```
Router> enable
Router# configure terminal
Router(config)# hw-module switch 2 module 1 logging onboard message level 7
Router(config)# end
```

## Related Commands

| Command | Description |
|---|---|
| **attach** | Connects to a specific line card for the purpose of executing commands on that card. |
| **clear logging onboard (Cat 6K)** | Clears onboard failure logs. |
| **copy logging onboard (Cat 6K)** | Copies OBFL data from the target OBFL-enabled module to a local or remote file system. |
| **show logging onboard (Cat 6K)** | Displays onboard failure logs. |

# show logging onboard (Cat 6K)

To display onboard failure logs (OBFL) on Cisco Catalyst 6000 series switches, use the **show logging onboard** command in privileged EXEC mode.

**show logging onboard** [**module** *module-number*] [**status** | [[**temperature** | **uptime** | **message**] [[**continuous** [**start** *start-time-and-date*] [**end** *end-time-and-date*]] | [**detail** [**start** *start-time-and-date*] [**end** *end-time-and-date*]] | [**summary**]]]]

## Syntax Description

| | |
|---|---|
| **module** *module-number* | (Optional) Specifies the module number. |
| **status** | (Optional) Displays the platform and CLI enable status for each of the test applications (system message, interrupt, temperature, and uptime). |
| **temperature** | (Optional) Displays temperature data. |
| **uptime** | (Optional) Displays system uptime data. |
| **message** | (Optional) Displays system messages collected at the level set by the **hw-module logging onboard** global configuration command. |
| **continuous** | (Optional) Can be used with the **message**, **temperature**, and **uptime** keywords to display continuously collected data. |
| **start** *start-time-and-date* **end** *end-time-and-date* | (Optional) Specifies a start and end time for **message**, **temperature**, and **uptime** reports. The **start** and **end** keywords can optionally be entered with the **continuous** and **detail** keywords.<br><br>The **start** and **end** keywords prompt for the time in 24-hour format (hh:mm:ss) followed by the date, the month in three-letter format (Jun for June, as an example), and the year in the range 1993 to 2035. Examples:<br><br>start 15:01:57 7 Mar 2007<br>end 15:04:57 14 Mar 2007 |
| **detail** | (Optional) Can be used with the **message**, **temperature**, and **uptime** keywords to display both summary and continuous data. |
| **summary** | (Optional) Displays summary data (default). |

## Command Modes

Privileged EXEC (#)

## Command History

| Release | Modification |
|---|---|
| 12.2(33)SXH | This command was introduced. |

## Usage Guidelines

The **show logging onboard** command can be entered without any arguments, which is the same as entering the **show logging onboard summary** command to display summarized information about OBFL for the device residing on the same module where the command is entered.

Use this command to view OBFL data from system hardware. The OBFL feature is enabled by default and records operating temperatures, hardware uptime, interrupts, and other important events and messages that can assist with diagnosing problems with hardware cards (or *modules*) installed in a Cisco

router or switch. Data is logged to files stored in nonvolatile memory. When the onboard hardware is started up, a first record is made for each area monitored and becomes a base value for subsequent records.

The OBFL feature provides a circular updating scheme for collecting continuous records and archiving older (historical) records, ensuring accurate data about the system. Data is recorded in one of two formats: continuous information that displays a snapshot of measurements and samples in a continuous file, and summary information that provides details about the data being collected. The message "No historical data to display" is seen when historical data is not available.

See the examples for more information about the type of data collected.

## Examples

### Temperature

Temperatures surrounding hardware modules can exceed recommended safe operating ranges and cause system problems such as packet drops. Higher than recommended operating temperatures can also accelerate component degradation and affect device reliability. Monitoring temperatures is important for maintaining environmental control and system reliability. Once a temperature sample is logged, the sample becomes the base value for the next record. From that point on, temperatures are recorded either when there are changes from the previous record or if the maximum storage time is exceeded. Temperatures are measured and recorded in degrees Celsius.

The following example shows how you might enter this command:

```
Router# show logging onboard module 2 temperature detail

--------------------------------------------------------------------------------
TEMPERATURE SUMMARY INFORMATION
--------------------------------------------------------------------------------
Number of sensors          : 12
Sampling frequency         : 5 minutes
Maximum time of storage    : 120 minutes
--------------------------------------------------------------------------------
Sensor                            |   ID   | Maximum Temperature 0C
--------------------------------------------------------------------------------
MB-Out                              980201     43
MB-In                               980202     28
MB                                  980203     29
MB                                  980204     38
EARL-Out                            910201     0
EARL-In                             910202     0
SSA 1                               980301     38
SSA 2                               980302     36
JANUS 1                             980303     36
JANUS 2                             980304     35
GEMINI 1                            980305     0
GEMINI 2                            980306     0
------------------------------------------------------------
Temp                         Sensor ID
0C     1    2    3    4    5    6    7    8    9   10   11   12
------------------------------------------------------------
No historical data to display
------------------------------------------------------------
--------------------------------------------------------------------------------
TEMPERATURE CONTINUOUS INFORMATION
--------------------------------------------------------------------------------
Sensor                            |   ID   |
--------------------------------------------------------------------------------
MB-Out                              980201
MB-In                               980202
MB                                  980203
```

```
MB                                      980204
EARL-Out                                910201
EARL-In                                 910202
SSA 1                                   980301
SSA 2                                   980302
JANUS 1                                 980303
JANUS 2                                 980304
GEMINI 1                                980305
GEMINI 2                                980306

-------------------------------------------------------------------------------
       Time Stamp    |Sensor Temperature 0C
MM/DD/YYYY HH:MM:SS |  1    2    3    4    5    6    7    8    9   10   11   12
-------------------------------------------------------------------------------
03/06/2007 22:32:51   31   26   27   27   NA   NA   33   32   30   29   NA   NA
03/06/2007 22:37:51   43   28   29   38   NA   NA   38   36   36   35   NA   NA
-------------------------------------------------------------------------------
```

Table 2 describes the significant fields shown in the display.

***Table 2 Temperature Summary Descriptions***

| Field | Description |
|---|---|
| Number of sensors | The total number of temperature sensors that will be recorded. A column for each sensor is displayed with temperatures listed under the number of each sensor, as available. |
| Sampling frequency | The time between measurements. |
| Maximum time of storage | Determines the maximum amount of time, in minutes, that can pass when the temperature remains unchanged and the data is not saved to storage media. After this time, a temperature record will be saved even if the temperature has not changed. |
| Sensor column | Lists the name of the sensor. |
| ID column | Lists an assigned identifier for the sensor. |
| Maximum Temperature 0C | Shows the highest recorded temperature per sensor. |
| Temp | Indicates a recorded temperature in degrees Celsius in the historical record. Columns following show the total time each sensor has recorded that temperature. |
| Sensor ID | An assigned number, so that temperatures for the same sensor can be stored together. |
| offset | Relative time of peer clock to local clock (in milliseconds). |
| disp | Dispersion |

### Operational Uptime

The operational uptime tracking begins when the module is powered on, and information is retained for the life of the module.

The following example shows how you might enter this command:

```
Router# show logging onboard module 2 uptime detail
```

```
--------------------------------------------------------------------------------
UPTIME SUMMARY INFORMATION
--------------------------------------------------------------------------------
First customer power on : 03/06/2007 22:32:51
Total uptime            :   0 years   0 weeks   2 days  18 hours  10 minutes
Total downtime          :   0 years   0 weeks   0 days   8 hours   7 minutes
Number of resets        : 130
Number of slot changes  : 16
Current reset reason    : 0xA1
Current reset timestamp : 03/07/2007 13:29:07
Current slot            : 2
Current uptime          :   0 years   0 weeks   1 days   7 hours   0 minutes
--------------------------------------------------------------------------------
Reset  |        |
Reason | Count  |
--------------------------------------------------------------------------------
0x5         64
0x6         62
0xA1         4
--------------------------------------------------------------------------------
--------------------------------------------------------------------------------
UPTIME CONTINUOUS INFORMATION
--------------------------------------------------------------------------------
Time Stamp          | Reset  | Uptime
MM/DD/YYYY HH:MM:SS | Reason | years weeks days hours  minutes
--------------------------------------------------------------------------------
03/06/2007 22:32:51   0xA1      0     0     0     0     0
--------------------------------------------------------------------------------
```

The operational uptime application tracks the following events:

- Date and time the customer first powered on a component.
- Total uptime and downtime for the component in years, weeks, days, hours, and minutes.
- Total number of component resets.
- Total number of slot (module) changes.
- Current reset timestamp to include the date and time.
- Current slot (module) number of the component.
- Current uptime in years, weeks, days, hours, and minutes.
- Reset reason; see Table 3 to translate the numbers displayed.
- Count is the number of resets that have occurred for each reset reason.

***Table 3 Reset Reason Codes and Explanations***

| Reset Reason Code (in hex) | Component/Explanation |
|---|---|
| 0x01 | Chassis on |
| 0x02 | Line card hot plug in |
| 0x03 | Supervisor requests line card off or on |
| 0x04 | Supervisor requests hard reset on line card |
| 0x05 | Line card requests Supervisor off or on |
| 0x06 | Line card requests hard reset on Supervisor |

*Table 3 Reset Reason Codes and Explanations (continued)*

| Reset Reason Code (in hex) | Component/Explanation |
|---|---|
| 0x07 | Line card self reset using the internal system register |
| 0x08 | — |
| 0x09 | — |
| 0x0A | Momentary power interruption on the line card |
| 0x0B | — |
| 0x0C | — |
| 0x0D | — |
| 0x0E | — |
| 0x0F | — |
| 0x10 | — |
| 0x11 | Off or on after Supervisor non-maskable interrupts (NMI) |
| 0x12 | Hard reset after Supervisor NMI |
| 0x13 | Soft reset after Supervisor NMI |
| 0x14 | — |
| 0x15 | Off or on after line card asks Supervisor NMI |
| 0x16 | Hard reset after line card asks Supervisor NMI |
| 0x17 | Soft reset after line card asks Supervisor NMI |
| 0x18 | — |
| 0x19 | Off or on after line card self NMI |
| 0x1A | Hard reset after line card self NMI |
| 0x1B | Soft reset after line card self NMI |
| 0x21 | Off or on after spurious NMI |
| 0x22 | Hard reset after spurious NMI |
| 0x23 | Soft reset after spurious NMI |
| 0x24 | — |
| 0x25 | Off or on after watchdog NMI |
| 0x26 | Hard reset after watchdog NMI |
| 0x27 | Soft reset after watchdog NMI |
| 0x28 | — |
| 0x29 | Off or on after parity NMI |
| 0x2A | Hard reset after parity NMI |
| 0x2B | Soft reset after parity NMI |
| 0x31 | Off or on after system fatal interrupt |
| 0x32 | Hard reset after system fatal interrupt |
| 0x33 | Soft reset after system fatal interrupt |

*Table 3 Reset Reason Codes and Explanations (continued)*

| Reset Reason Code (in hex) | Component/Explanation |
|---|---|
| 0x34 | — |
| 0x35 | Off or on after application-specific integrated circuit (ASIC) interrupt |
| 0x36 | Hard reset after ASIC interrupt |
| 0x37 | Soft reset after ASIC interrupt |
| 0x38 | — |
| 0x39 | Off or on after unknown interrupt |
| 0x3A | Hard reset after unknown interrupt |
| 0x3B | Soft reset after unknown interrupt |
| 0x41 | Off or on after CPU exception |
| 0x42 | Hard reset after CPU exception |
| 0x43 | Soft reset after CPU exception |
| 0xA1 | Reset data converted to generic data |

### Interrupts

Interrupts are generated by system components that require attention from the CPU, such as ASICs and NMIs. Interrupts are generally related to hardware limit conditions or errors that need to be corrected.

The continuous format records each time a component is interrupted, and this record is stored and used as base information for subsequent records. Each time the list is saved, a timestamp is added. Time differences from the previous interrupt are counted, so that technical personnel can gain a complete record of the component's operational history when an error occurs.

The following example shows how you might enter this command:

```
Router# show logging onboard module 2 interrupt detail

--------------------------------------------------------------------------------
INTERRUPT SUMMARY INFORMATION
--------------------------------------------------------------------------------
Name                                              |  ID | Offset | Bit |  Count
--------------------------------------------------------------------------------
No historical data to display
--------------------------------------------------------------------------------
--------------------------------------------------------------------------------
CONTINUOUS INTERRUPT INFORMATION
--------------------------------------------------------------------------------
MM/DD/YYYY HH:MM:SS mmm | Name                                |  ID | Offset | Bit
--------------------------------------------------------------------------------
03/06/2007 22:33:06 450   Port-ASIC #2                          9   0x00E7     6
--------------------------------------------------------------------------------
```

Table 4 describes the significant fields shown in the display.

*Table 4 Interrupt Summary Information*

| Field | Description |
|---|---|
| Name | A description of the component including its position in the device. |
| ID | An assigned field for data storage. |
| Offset | The location of the next block in bytes. |
| Bit | The interrupt bit number recorded from the component's internal register. |
| The timestamp | Shows the date and time that an interrupt occurred to the millisecond. |

### Message Logging

The OBFL feature logs standard system messages. Instead of displaying the message to a terminal, the message is written to and stored in a file, so the message can be accessed and read at a later time. System messages range from level 1 alerts to level 7 debug messages, and these levels can be specified in the **hw module logging onboard** command.

The following example shows how you might enter this command:

```
Router# show logging onboard module 2 message detail

--------------------------------------------------------------------------------
ERROR MESSAGE SUMMARY INFORMATION
--------------------------------------------------------------------------------
Facility-Sev-Name      | Count | Persistence Flag
MM/DD/YYYY HH:MM:SS
--------------------------------------------------------------------------------
No historical data to display
--------------------------------------------------------------------------------
--------------------------------------------------------------------------------
ERROR MESSAGE CONTINUOUS INFORMATION
--------------------------------------------------------------------------------
MM/DD/YYYY HH:MM:SS Facility-Sev-Name
--------------------------------------------------------------------------------
03/06/2007 22:33:35  %GOLD_OBFL-3-GOLD : Diagnostic OBFL: Diagnostic OBFL testing
```

Table 5 describes the significant fields shown in the display.

*Table 5 Error Message Summary Information*

| Field | Description |
|---|---|
| A timestamp | Shows the date and time the message was logged. |
| Facility-Sev-Name | A coded naming scheme for a system message, as follows:<br>• The Facility code consists of two or more uppercase letters that indicate the hardware device (facility) to which the message refers.<br>• Sev is a single-digit code from 1 to 7 that reflects the severity of the message.<br>• Name is one or two code names separated by a hyphen that describe the part of the system from where the message is coming. |

*Table 5 Error Message Summary Information (continued)*

| Field | Description |
|---|---|
| Error message | Follows the Facility-Sev-Name codes. For more information about system messages, see the *Cisco IOS System and Error Messages* guide. |
| Count | Indicates the number of instances of this message that is allowed in the history file. Once that number of instances has been recorded, the oldest instance will be removed from the history file to make room for new ones. |
| Persistence Flag | Gives a message priority over others that do not have the flag set. |

## Related Commands

| Command | Description |
|---|---|
| **attach** | Connects to a specific line card for the purpose of executing commands on that card. |
| **clear logging onboard (Cat 6K)** | Clears onboard failure logs. |
| **copy logging onboard (Cat 6K)** | Copies OBFL data from the target OBFL-enabled module to a local or remote file system. |
| **hw-module logging onboard (Cat 6K)** | Disables and enables OBFL. |

# Feature Information for OBFL

Table 6 lists the release history for this feature.

Not all commands may be available in your Cisco IOS software release. For release information about a specific command, see the command reference documentation.

Use Cisco Feature Navigator to find information about platform support and software image support. Cisco Feature Navigator enables you to determine which Cisco IOS and Catalyst OS software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

**Note** Table 6 lists only the Cisco IOS software release that introduced support for a given feature in a given Cisco IOS software release train. Unless noted otherwise, subsequent releases of that Cisco IOS software release train also support that feature.

*Table 6 Feature Information for OBFL*

| Feature Name | Releases | Feature Information |
|---|---|---|
| Onboard Failure Logging | 12.2(33)SXH | The Onboard Failure Logging (OBFL) feature collects data such as operating temperatures, hardware uptime, interrupts, and other important events and messages from system hardware installed in a Cisco router or switch. The data is stored in nonvolatile memory and helps technical personnel diagnose hardware problems.<br><br>In Release 12.2(33)SXH, this feature was introduced on the Cisco Catalyst 6000 series switches.<br><br>The following commands were introduced for the Cisco Catalyst 6000 series switches by this feature: **clear logging onboard** (Cat 6K), **copy logging onboard module** (Cat 6K), **hw-module logging onboard** (Cat 6K), **show logging onboard** (Cat 6K). |