



# Trace Management

---

The following sections are included in this chapter:

- [Tracing Overview, on page 1](#)
- [How Tracing Works, on page 1](#)
- [Tracing Levels, on page 2](#)
- [Viewing a Tracing Level, on page 3](#)
- [Setting a Tracing Level, on page 5](#)
- [Viewing the Content of the Trace Buffer, on page 5](#)

## Tracing Overview

Tracing is a function that logs internal events. Trace files containing trace messages are automatically created and saved to the `tracelogs` directory on the hard disk: file system on the router, which stores tracing files in bootflash.

The contents of trace files are useful for the following purposes:

- **Troubleshooting**—Helps to locate and solve an issue with a router. The trace files can be accessed in diagnostic mode even if other system issues are occurring simultaneously.
- **Debugging**—Helps to obtain a detailed view of system actions and operations.

## How Tracing Works

Tracing logs the contents of internal events on a router. Trace files containing all the trace output pertaining to a module are periodically created and updated and stored in the `tracelog` directory. Trace files can be erased from this directory to recover space on the file system without impacting system performance. The files can be copied to other destinations using file transfer functions (such as FTP and TFTP) and opened using a plain text editor.



---

**Note** Tracing cannot be disabled on a router.

---

Use the following commands to view trace information and set tracing levels:

- **show platform software trace message**—Shows the most recent trace information for a specific module. This command can be used in privileged EXEC and diagnostic modes. When used in diagnostic mode, this command can gather trace log information during a Cisco IOS XE failure.
- **set platform software trace**—Sets a tracing level that determines the types of messages that are stored in the output. For more information on tracing levels, see [Tracing Levels, on page 2](#).

## Tracing Levels

Tracing levels determine how much information should be stored about a module in the trace buffer or file.

The following table shows all the tracing levels that are available and provides descriptions of what types of messages are displayed with each tracing level.

**Table 1: Tracing Levels and Descriptions**

Tracing Level	Level Number	Description
Emergency	0	The message is regarding an issue that makes the system unusable.
Alert	1	The message is regarding an action that must be taken immediately.
Critical	2	The message is regarding a critical condition. This is the default setting for every module on the router.
Error	3	The message is regarding a system error.
Warning	4	The message is regarding a system warning.
Notice	5	The message is regarding a significant issue, but the router is still working normally.
Informational	6	The message is useful for informational purposes only.
Debug	7	The message provides debug-level output.
Verbose	8	All possible tracing messages are sent.

Tracing Level	Level Number	Description
Noise	—	All possible trace messages pertaining to a module are logged.  The noise level is always equal to the highest possible tracing level. Even if a future enhancement to tracing introduces a higher tracing level than verbose level, the noise level will become equal to the level of the newly introduced tracing level.

If a tracing level is set, messages are collected from both lower tracing levels and from its own level.

For example, setting the tracing level to 3 (error) means that the trace file will contain output messages for levels: 0 (emergencies), 1 (alerts), 2 (critical), and 3 (error).

If you set the trace level to 4 (warning), it results in output messages for levels: 0 (emergencies), 1 (alerts), 2 (critical), 3 (error), and 4 (warning).

The default tracing level for every module on the router is 5 (notice).

A tracing level is not set in a configuration mode, which results in tracing-level settings being returned to default values after the router reloads.



**Caution** Setting the tracing level of a module to debug level or higher can have a negative impact on the performance.



**Caution** Setting high tracing levels on a large number of modules can severely degrade performance. If a high tracing level is required in a specific context, it is almost always preferable to set the tracing level of a single module to a higher level rather than setting multiple modules to high levels.

## Viewing a Tracing Level

By default, all the modules on a router are set to 5 (notice). This setting is maintained unless changed by a user.

To see the tracing level for a module on a router, enter the **show platform software trace level** command in privileged EXEC mode or diagnostic mode.

The following example shows how the **show platform software trace level** command is used to view the tracing levels of the forwarding manager processes on an active RP:

```
Router# show platform software trace level forwarding-manager rp active
Module Name                               Trace Level
-----
acl                                         Notice
binos                                       Notice
binos/brand                               Notice
bipc                                       Notice
```

bsignal	Notice
btrace	Notice
cce	Notice
cdllib	Notice
cef	Notice
chasfs	Notice
chasutil	Notice
erspan	Notice
ess	Notice
ether-channel	Notice
evlib	Notice
evutil	Notice
file_alloc	Notice
fman_rp	Notice
fpm	Notice
fw	Notice
icmp	Notice
interfaces	Notice
iosd	Notice
ipc	Notice
ipclog	Notice
iphc	Notice
IPsec	Notice
mgmte-acl	Notice
mlp	Notice
mqipc	Notice
nat	Notice
nbar	Notice
netflow	Notice
om	Notice
peer	Notice
qos	Notice
route-map	Notice
sbc	Notice
services	Notice
sw_wdog	Notice
tcl_acl_config_type	Notice
tcl_acl_db_type	Notice
tcl_cdlcore_message	Notice
tcl_cef_config_common_type	Notice
tcl_cef_config_type	Notice
tcl_dpiddb_config_type	Notice
tcl_fman_rp_comm_type	Notice
tcl_fman_rp_message	Notice
tcl_fw_config_type	Notice
tcl_hapi_tcl_type	Notice
tcl_icmp_type	Notice
tcl_ip_options_type	Notice
tcl_ipc_ack_type	Notice
tcl_IPsec_db_type	Notice
tcl_mcp_comm_type	Notice
tcl_mlp_config_type	Notice
tcl_mlp_db_type	Notice
tcl_om_type	Notice
tcl_ui_message	Notice
tcl_ui_type	Notice
tcl_urpf_config_type	Notice
tdllib	Notice
trans_avl	Notice
uihandler	Notice
uipeer	Notice
uistatus	Notice
urpf	Notice
vista	Notice

wccp

Notice

## Setting a Tracing Level

To set a tracing level for a module on a router, or for all the modules within a process on a router, enter the **set platform software trace** command in the privileged EXEC mode or diagnostic mode.

The following example shows the tracing level for the ACL module in the Forwarding Manager of the ESP processor in slot 0 set to `info`:

```
set platform software trace forwarding-manager F0 acl info
```

## Viewing the Content of the Trace Buffer

To view the trace messages in the trace buffer or file, enter the **show platform software trace message** command in privileged EXEC or diagnostic mode. In the following example, the trace messages for the Host Manager process in Route Processor slot 0 are viewed using the **show platform software trace message** command:

```
Router# show platform software trace message host-manager R0
08/23 12:09:14.408 [uipeer]: (info): Looking for a ui_req msg
08/23 12:09:14.408 [uipeer]: (info): Start of request handling for con 0x100a61c8
08/23 12:09:14.399 [uipeer]: (info): Accepted connection for 14 as 0x100a61c8
08/23 12:09:14.399 [uipeer]: (info): Received new connection 0x100a61c8 on descriptor 14
08/23 12:09:14.398 [uipeer]: (info): Accepting command connection on listen fd 7
08/23 11:53:57.440 [uipeer]: (info): Going to send a status update to the shell manager in
slot 0
08/23 11:53:47.417 [uipeer]: (info): Going to send a status update to the shell manager in
slot 0
```

