



## CHAPTER 8

# Log and Debug Commands for DPE

---

This chapter describes the command line interface (CLI) commands that you can use to debug the Cisco Broadband Access Center (Cisco BAC) Device Provisioning Engine (DPE), and monitor and manage the Cisco BAC log system.



### Note

Before using any debug command, ensure that DPE debugging is enabled. Run the **debug on** command to enable this function. See [debug on, page 8-5](#), for more information.

---

The commands described in this section are:

- [clear logs, page 8-2](#)
- [debug dpe, page 8-2](#)
  - [debug dpe cache, page 8-2](#)
  - [debug dpe chatty-client, page 8-2](#)
  - [debug dpe connection, page 8-3](#)
  - [debug dpe dpe-ext, page 8-3](#)
  - [debug dpe dpe-server, page 8-3](#)
  - [debug dpe event-manager, page 8-3](#)
  - [debug dpe exceptions, page 8-4](#)
  - [debug dpe framework, page 8-4](#)
  - [debug dpe messaging, page 8-4](#)
  - [debug dpe statistics, page 8-4](#)
- [debug on, page 8-5](#)
- [no debug, page 8-5](#)
- [log level, page 8-5](#)
- [show log, page 8-6](#)

## clear logs

Use this command to remove historic (out-of-date) log files that exist on the system. These files include:

- DPE logs
- Syslog

Over time, historic log files accumulate in the DPE. The **support bundle state** command is used to bundle these logs. We recommend that you create a bundle before clearing logs to ensure that no necessary files are accidentally lost.

After you enter this command, prompts appear to indicate that logs are being cleared. The number of log files that are cleared, is also identified.

### Examples

```
dpe# clear logs
Clearing historic log files...
+ Removing 1 DPE log files...
+ No more historic logs.
```

## debug dpe

The **debug dpe** is the global syntax of the commands that you use to debug the various services on the DPE.



### Note

If you run the following commands on an unlicensed DPE, a message similar to this one appears:

```
This DPE is not licensed. Your request cannot be serviced.
Please check with your system administrator for DPE licenses.
```

[Table 8-1](#) describes the various commands you can use to debug the DPE.

**Table 8-1** List of debug dpe Commands

Command Usage	Example
<b>debug dpe cache</b>	
<b>no debug dpe cache</b>	
Enables you to debug DPE cache logging, which involves messages pertaining to the DPE cache including: <ul style="list-style-type: none"> <li>• Logging requests for cache entries.</li> <li>• Updates to the cache.</li> <li>• Other interactions by DPE subsystems.</li> </ul> To disable DPE cache debug logging, use the <b>no</b> form of this command.	dpe# <b>debug dpe cache</b> % OK
<b>debug dpe chatty-client</b>	
<b>no debug dpe chatty-client</b>	

**Table 8-1** List of debug dpe Commands (continued)

Command Usage	Example
<p>Enables you to debug the chatty-client service, which logs chatty-client service status and error messages.</p> <p>To disable the debugging of the chatty-client service, use the <b>no</b> form of this command.</p>	<pre>dpe# <b>debug dpe chatty-client</b> % OK</pre>
<b>debug dpe connection</b>	
<b>no debug dpe connection</b>	
<p>Enables you to debug the DPE connection, which logs communication subsystem status and error messages. Use this command for finding communication problems between the DPE and the RDU.</p> <p>To disable the debugging of the DPE connection, use the <b>no</b> form of this command.</p>	<pre>dpe# <b>debug dpe connection</b> % OK</pre>
<b>debug dpe dpe-ext</b>	
<b>no debug dpe dpe-ext</b>	
<p>Enables you to debug the DPE extensions, which involves logging messages about the overall status and issues of the DPE extensions.</p> <p>To disable debugging of the DPE extensions, use the <b>no</b> form of this command.</p>	<pre>dpe# <b>debug dpe dpe-ext</b> % OK</pre>
<b>debug dpe dpe-server</b>	
<b>no debug dpe dpe-server</b>	
<p>Enables you to debug the DPE server, which involves logging messages about the overall status and issues of the DPE server.</p> <p>To disable debugging of the DPE server, use the <b>no</b> form of this command.</p>	<pre>dpe# <b>debug dpe dpe-server</b> % OK</pre>
<b>debug dpe event-manager</b>	
<b>no debug dpe event-manager</b>	
<p>Enables you to debug the DPE event manager, which involves logging messages and conditions showing the state of the event manager.</p> <p>To disable debugging of the DPE event manager, use the <b>no</b> form of this command.</p> <p>Debugging of the DPE event manager is, by default, enabled.</p>	<pre>dpe# <b>debug dpe event-manager</b> % OK</pre>

Table 8-1 List of debug dpe Commands (continued)

Command Usage	Example
<b>debug dpe exceptions</b>	
<b>no debug dpe exceptions</b>	
<p>Enables you to debug the DPE exceptions, which involves logging full stack traces for exceptions occurring during system operation.</p> <p>When unusual situations occur, where the system is apparently corrupt or behaving abnormally, running this command can reveal valuable information for the Cisco TAC support.</p> <p>To disable the debugging of DPE exceptions, use the <b>no</b> form of this command.</p> <p>Debugging of DPE exceptions is, by default, enabled.</p>	<pre>dpe# debug dpe exceptions % OK</pre>
<b>debug dpe framework</b>	
<b>no debug dpe framework</b>	
<p>Enables you to debug the DPE framework, which involves logging information about the DPE server's underlying framework. This underlying infrastructure provides support for all of the various servers in Cisco BAC.</p> <p>To disable the debugging of the DPE framework, use the <b>no</b> form of this command.</p> <p>Debugging of the DPE framework is, by default, enabled.</p>	<pre>dpe# debug dpe framework % OK</pre>
<b>debug dpe messaging</b>	
<b>no debug dpe messaging</b>	
<p>Enables you to debug the DPE messaging, which involves logging details about the DPE messaging subsystem. This subsystem is used primarily for communication between the DPE and the RDU.</p> <p>To disable the debugging of DPE messaging, use the <b>no</b> form of this command.</p>	<pre>dpe# debug dpe messaging % OK</pre>
<b>debug dpe statistics</b>	
<b>no debug dpe statistics</b>	
<p>Enables you to collect the performance statistics.</p> <p>To disable debugging of the DPE performance statistics collection, use the <b>no</b> form of this command.</p>	<pre>dpe# debug dpe statistics % OK</pre>

## debug on

Use this command to enable debug logging, which can be helpful when troubleshooting possible system problems. Additionally, specific debugging categories must be enabled separately with commands such as **debug dpe cache**.

To disable debug logging, run the **no debug** command. See [no debug, page 8-5](#), for more information.



### Caution

Enabling debug logging may have a severe impact on DPE performance. The DPE should never be left running with debug turned on for long periods of time.

If you run this command on an unlicensed DPE, a message similar to this one appears:

```
This DPE is not licensed. Your request cannot be serviced.  
Please check with your system administrator for DPE licenses.
```

### Defaults

Debug logging is, by default, enabled.

### Examples

```
dpe# debug on  
% OK
```

## no debug

Use this command to disable all debug logging.

If you run this command on an unlicensed DPE, a message similar to this one appears:

```
This DPE is not licensed. Your request cannot be serviced.  
Please check with your system administrator for DPE licenses.
```

To enable debugging, use the **debug on** command. For more information, see [debug on, page 8-5](#).

### Examples

```
dpe# no debug  
% OK
```

## log level

Use this command to set the level of minimum DPE log messages that will be saved, as described in the *Cisco Broadband Access Center Administrator's Guide, Release 3.5, 3.5.1, 3.5.2*.

If you run this command on an unlicensed DPE, a message similar to this one appears:

```
This DPE is not licensed. Your request cannot be serviced.  
Please check with your system administrator for DPE licenses.
```

**Syntax Description**

`log level number`

*number*—Identifies the logging level, by number, to be saved. The log levels that Cisco BAC supports are described in [Table 8-2](#).

**Table 8-2 DPE Log Levels**

Log Level No.	Description
0-emergency	Saves all emergency messages
1-alert	Saves all activities that need immediate action and those of a more severe nature
2-critical	Saves all critical conditions and those of a more severe nature
3-error	Saves all error messages and those of a more severe nature
4-warning	Saves all warning messages and those of a more severe nature
5-notification	Saves all notification messages and those of a more severe nature
6-info	Saves all logging messages available

Setting a specific log level saves messages less than or equal to the configured level. For example, when you set the log level at 5-notification, all events generating messages with a log level of 4 or less are written into the log file.

The logging system's log levels are used to identify the urgency with which you might want to address log issues. The 0-emergency setting is the most severe level of logging while 6-info is the least severe, saving mostly informational log messages.

**Defaults**

The level of minimum DPE log messages that will be saved is, by default, set at 5-notification.

**Examples**

```
dpe# log level 6
% OK
```

# show log

Use this command to show all recent log entries for the DPE. These logs contain general DPE process information, including logging all system errors or severe problems. Check this log when the system is experiencing difficulties. If the log contains insufficient information, enable the debug logging function and experiment with the different categories related to the problem.

**Syntax Description**

`show log [last 1..999 | run]`

- **last 1..999**—Shows the specified number of recent log entries for the DPE, with *1..999* specifying the number of log entries that you want to display. This element is optional.
- **run**—Displays the running DPE log, which starts showing all messages logged to the DPE log. The command continues to run until you press Enter. This element is optional.

---

**Examples****Example 1**

```
dpe# show log
2006 02 14 07:50:26 EST: %BAC-DPE-7-DEBUG_FRAMEWORK: ThreadMonitor:
BACThread[Connector,5,BAC,alive]
```



---

**Note** The output of this command has been shortened for demonstration purposes.

---

**Example 2**

```
dpe# show log last 3
2006 02 14 07:51:26 EST: %BAC-DPE-7-DEBUG_FRAMEWORK: ThreadMonitor:      Cwmp1Thread-1
2006 02 14 07:51:26 EST: %BAC-DPE-7-DEBUG_FRAMEWORK: ThreadMonitor:      Http1Thread-0
2006 02 14 07:51:26 EST: %BAC-DPE-7-DEBUG_FRAMEWORK: ThreadMonitor:      Http1Thread-1
```

**Example 3**

```
dpe# show log run
% Press <enter> to stop.
2006 02 14 07:53:22 EST: %BAC-DPE-7-DEBUG_FRAMEWORK: OSStatusService: current CPU load
percentage 1%
2006 02 14 07:53:25 EST: %BAC-DPE-7-DEBUG_FRAMEWORK: MemoryMonitor: Memory:
2006 02 14 07:53:25 EST: %BAC-DPE-7-DEBUG_FRAMEWORK: MemoryMonitor: Total memory 29777920
2006 02 14 07:53:25 EST: %BAC-DPE-7-DEBUG_FRAMEWORK: MemoryMonitor: Free memory 4058120
2006 02 14 07:53:26 EST: %BAC-DPE-7-DEBUG_FRAMEWORK: ThreadMonitor: Threads:

Stopped.
```

■ show log