



# DPE 用のログおよびデバッグ コマンド

この章では、Broadband Access Center (BAC) の Device Provisioning Engine (DPE) をデバッグし、BAC ログ システムを管理および監視するために使用するコマンドライン インターフェイス (CLI) コマンドについて説明します。



(注)

任意のデバッグ コマンドを使用する前に、DPE のデバッグがイネーブルになっていることを確認してください。この機能をイネーブルにするには、**debug on** コマンドを実行します。詳細については、P.6-5 の「[debug on](#)」を参照してください。

この項で説明するコマンドは、次のとおりです。

- [clear logs](#) (P.6-2)
- [debug dpe](#) (P.6-3)
  - [debug dpe cache](#) (P.6-3)
  - [debug dpe connection](#) (P.6-3)
  - [debug dpe dpe-server](#) (P.6-3)
  - [debug dpe event-manager](#) (P.6-4)
  - [debug dpe exceptions](#) (P.6-4)
  - [debug dpe framework](#) (P.6-4)
  - [debug dpe messaging](#) (P.6-4)
  - [debug dpe statistics](#) (P.6-4)
- [debug on](#) (P.6-5)
- [no debug](#) (P.6-5)
- [log level](#) (P.6-6)
- [show log](#) (P.6-7)

## clear logs

このコマンドは、システムに存在する古いログ ファイルを削除するときに使用します。次のファイルが対象となります。

- DPE ログ
- Syslog

時の経過とともに、古いログ ファイルは DPE 内に蓄積します。このようなログをバンドルするには、**support bundle state** コマンドを使用します。必要なファイルが消失されるのを防ぐため、ログをクリアする前に、バンドルを作成することをお勧めします。

このコマンドを使用すると、ログをクリアしていることを示すプロンプトが表示されます。クリアされたログ ファイルの数も示されます。

---

### 例

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

# debug dpe

**debug dpe** は、DPE のさまざまなサービスのデバッグに使用するコマンドのグローバル構文です。



(注)

次の表に示すコマンドをライセンスのない DPE で実行した場合は、次のようなメッセージが表示されます。

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

表 6-1 は、DPE のデバッグで使用できるさまざまなコマンドを示します。

表 6-1 debug dpe コマンドのリスト

コマンドの使用方法	例
<b>debug dpe cache</b>  <b>no debug dpe cache</b>  DPE キャッシュのデバッグ ロギングをイネーブルにします。次のような DPE キャッシュに関するメッセージが記録されます。 <ul style="list-style-type: none"> <li>• キャッシュ エントリのロギングの要求。</li> <li>• キャッシュのアップデート。</li> <li>• DPE サブシステムによるその他のインタラクション。</li> </ul> DPE キャッシュのデバッグ ロギングをディセーブルにするには、このコマンドの <b>no</b> 形式を使用します。	<pre>dpe# debug dpe cache % OK</pre>
<b>debug dpe connection</b>  <b>no debug dpe connection</b>  DPE 接続のデバッグをイネーブルにします。通信サブシステムのステータスやエラー メッセージが記録されます。このコマンドは、DPE と RDU との間の通信に関する問題を検出するときに使用します。DPE 接続のデバッグをディセーブルにするには、このコマンドの <b>no</b> 形式を使用します。	<pre>dpe# debug dpe connection % OK</pre>
<b>debug dpe dpe-server</b>  <b>no debug dpe dpe-server</b>  DPE サーバのデバッグをイネーブルにします。DPE サーバの総合的なステータスや問題に関するログ メッセージが記録されます。DPE サーバのデバッグをディセーブルにするには、このコマンドの <b>no</b> 形式を使用します。	<pre>dpe# debug dpe dpe-server % OK</pre>

表 6-1 debug dpe コマンドのリスト (続き)

コマンドの使用方法	例
<p><b>debug dpe event-manager</b></p> <p><b>no debug dpe event-manager</b></p> <p>DPE イベント マネージャのデバッグをイネーブルにします。イベント マネージャのステータスに関するログ メッセージや状態が記録されます。DPE イベント マネージャのデバッグをディセーブルにするには、このコマンドの <b>no</b> 形式を使用します。</p> <p>DPE イベント マネージャのデバッグは、デフォルトではイネーブルになっています。</p>	<pre>dpe# debug dpe event-manager % OK</pre>
<p><b>debug dpe exceptions</b></p> <p><b>no debug dpe exceptions</b></p> <p>DPE 例外のデバッグをイネーブルにします。システムのエラー発生中に発生した例外の完全なスタック トレースが記録されます。システムの破損や異常動作のように異例の事態が発生した場合、Cisco TAC サポートにお問い合わせいただく前にこのコマンドを実行すると、貴重な情報を提示することができます。DPE 例外のデバッグをディセーブルにするには、このコマンドの <b>no</b> 形式を使用します。</p> <p>DPE 例外のデバッグは、デフォルトではイネーブルになっています。</p>	<pre>dpe# debug dpe exceptions % OK</pre>
<p><b>debug dpe framework</b></p> <p><b>no debug dpe framework</b></p> <p>DPE フレームワークのデバッグをイネーブルにします。DPE サーバの基礎をなすフレームワークに関するログ情報が記録されます。この基礎をなすインフラストラクチャによって、BAC の各種サーバは支えられています。DPE フレームワークのデバッグをディセーブルにするには、このコマンドの <b>no</b> 形式を使用します。</p> <p>DPE フレームワークのデバッグは、デフォルトではイネーブルになっています。</p>	<pre>dpe# debug dpe framework % OK</pre>
<p><b>debug dpe messaging</b></p> <p><b>no debug dpe messaging</b></p> <p>DPE メッセージングのデバッグをイネーブルにします。DPE のメッセージング サブシステムに関する詳細情報が記録されます。このサブシステムは、主に DPE と RDU との間の通信に使用されます。DPE メッセージングのデバッグをディセーブルにするには、このコマンドの <b>no</b> 形式を使用します。</p>	<pre>dpe# debug dpe messaging % OK</pre>
<p><b>debug dpe statistics</b></p> <p><b>no debug dpe statistics</b></p> <p>パフォーマンス統計の集合をイネーブルにします。DPE パフォーマンス統計の集合のデバッグをディセーブルにするには、このコマンドの <b>no</b> 形式を使用します。</p>	<pre>dpe# debug dpe statistics % OK</pre>

## debug on

このコマンドは、デバッグ ロギングをイネーブルにするときに使用します。予想されるシステムの問題をトラブルシューティングするときに役立ちます。このコマンド以外に、**debug dpe cache**などのコマンドを使用して、特定のデバッグ カテゴリを個々にイネーブルにする必要があります。

デバッグ ロギングをディセーブルにするには、**no debug** コマンドを実行します。詳細については、[P.6-5](#)の「**no debug**」を参照してください。



### 注意

デバッグ ロギングをイネーブルにすると、DPE のパフォーマンスに重大な影響が及ぶ可能性があります。デバッグをイネーブルにした状態で、DPE を長時間にわたって実行しないようにします。

このコマンドをライセンスのない DPE で実行した場合は、次のようなメッセージが表示されます。

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

### デフォルト

デバッグ ロギングは、デフォルトではイネーブルになっています。

### 例

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

## no debug

このコマンドは、すべてのデバッグ ロギングをディセーブルにするときに使用します。

このコマンドをライセンスのない DPE で実行した場合は、次のようなメッセージが表示されます。

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

デバッグをイネーブルにするには、**debug on** コマンドを使用します。詳細については、[P.6-5](#)の「**debug on**」を参照してください。

### 例

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

# log level

このコマンドは、保存される DPE ログ メッセージの最小レベルを設定するときに使用します。ログレベルの詳細については、『Cisco Broadband Access Center Administrator's Guide, Release 3.0』を参照してください。

このコマンドをライセンスのない DPE で実行した場合は、次のようなメッセージが表示されます。

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

## シンタックスの説明

`log level number`

*number* : 保存されるログ レベルを数字で表します。BAC がサポートするログ レベルは、表 6-2 のとおりです。

表 6-2 DPE ログ レベル

ログ レベル番号	説明
0 (緊急)	緊急なメッセージをすべて保存します。
1 (アラート)	即時のアクションが必要なアクティビティ、およびそれ以上のレベルのアクティビティをすべて保存します。
2 (クリティカル)	異常な状態、およびそれ以上のレベルの状態をすべて保存します。
3 (エラー)	エラー メッセージ、およびそれ以上のレベルのメッセージをすべて保存します。
4 (警告)	警告メッセージ、およびそれ以上のレベルのメッセージをすべて保存します。
5 (通知)	通知メッセージ、およびそれ以上のレベルのメッセージをすべて保存します。
6 (情報)	出力されたログ メッセージをすべて保存します。



(注) 特定のログ レベルを設定すると、設定されたレベルおよびそれ以下のメッセージが保存されます。たとえば、ログ レベルを 5 (通知) に設定すると、レベル 4 以下のログ レベルのメッセージを生成するイベントがすべてログ ファイルに書き込まれます。ロギング システムのログ レベルは、ログの問題に対処する際の緊急性を表すために使用されます。ログ レベルの中では、0 (緊急) が最も重要度が高いレベルで、主に情報ログ メッセージを保存する 6 (情報) が最も重要度が低いレベルです。

## デフォルト

保存される DPE ログ メッセージの最小レベルは、デフォルトでは 5 (通知) に設定されています。

## 例

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

# show log

このコマンドは、DPEの最新のログ エントリをすべて表示するときに使用します。ログには、システム エラーや重大な問題のロギングを含め、一般的な DPE プロセスの情報が記録されます。システムが困難な状態に陥っているときは、このログをチェックします。ログに含まれる情報が不十分な場合は、デバッグ ロギング機能をイネーブルにして、問題に関連したカテゴリをさまざまに変更してみます。

## シンタックスの説明

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

- **last 1..999** : DPE の最新のログ エントリから、指定された数のエントリを表示します。1..999 は、表示するログ エントリの数を指定しますこれは省略可能な要素です。
- **run** : 実行中の DPE ログを表示します。このコマンドを実行すると、DPE ログに記録されたすべてのメッセージの表示が開始されます。コマンドの実行は、Enter キーを押すまで継続されません。これは省略可能な要素です。

## 例

### 例 1

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



(注) ここでは、説明のためにコマンドの出力例の一部のみ紹介しています。

### 例 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
```

### 例 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