Cisco Prime Cable Provisioning 6.1.3 DPE CLI Reference Guide

September 17, 2019

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

Text Part Number:

Cisco Prime Cable Provisioning 6.1.3 DPE CLI Reference Guide

CONTENTS

Preface

The *Cisco Prime Cable Provisioning 6.1.3 DPE CLI Reference Guide,* describes the command-line interface (CLI) commands that this release of Cisco Prime Cable Provisioning, which is called Prime Cable Provisioning throughout the guide, supports on the Device Provisioning Engine (DPE).

Audience

This guide is written for those using the CLI of the Prime Cable Provisioning DPE.

Product Documentation

Note We sometimes update the printed and electronic documentation after original publication. Therefore, you should also review the documentation on Cisco.com for any updates.

See the *Cisco Prime Cable Provisioning 6.1.3 Documentation Overview* for the list of Prime Cable Provisioning guides.

Related Documentation

See the *Cisco Prime Network Registrar 10.x Documentation Overview* for the list of Cisco Prime Network Registrar guides.

Communications, Services, and Additional Information

- To receive timely, relevant information from Cisco, sign up at Cisco Profile Manager.
- To get the business impact you're looking for with the technologies that matter, visit Cisco Services.
- To submit a service request, visit Cisco Support.
- To discover and browse secure, validated enterprise-class apps, products, solutions and services, visit Cisco Marketplace.
- To obtain general networking, training, and certification titles, visit Cisco Press.
- To find warranty information for a specific product or product family, access Cisco Warranty Finder.

Cisco Bug Search Tool

Cisco Bug Search Tool (BST) is a web-based tool that acts as a gateway to the Cisco bug tracking system that maintains a comprehensive list of defects and vulnerabilities in Cisco products and software. BST provides you with detailed defect information about your products and software.

Introduction to DPE CLI

This chapter describes licensing and authentication requirements for the Cisco Prime Cable Provisioning Device Provisioning Engine (DPE) and how you can access the command-line interface (CLI) of the DPE.

DPE Licensing

Licensing controls the number of DPEs that you can use. To configure the DPE from the CLI, you must have a valid license. If you run the commands described in this guide on an unlicensed DPE, the following message appears:

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

For details on how to obtain the license file, see the *Cisco Prime Cable Provisioning 6.1.3 User Guide*.

Once you receive your license file, install Prime Cable Provisioning. Then, from the Admin UI, use the following procedure to install the licenses that you purchased:

Note Before installing your license, ensure that you back it up in case you have to reinstall Prime Cable Provisioning.

Step 1 Once you receive your license file, save each file on the system from which you intend to launch the Prime Cable Provisioning Admin UI.

Step 2 Launch your web browser on that system.

Step 3 Enter the administrator's location using this syntax:

`https://machine_name:port_number/`

- *machine_name*—Identifies the computer on which the RDU is running.
- *port_number*—Identifies the computer port on which the server side of the administrator application runs.

The default port number is:

- 8100 for HTTP over TCP
- 8443 for HTTP over SSL

The main login page appears.

Step 4 Enter the default username (**admin**) and default password (**changeme**).

Note If you are logging in for the first time, the Change Password screen appears. Enter a new password and confirm it. The password that you enter must have at least eight characters.

Step 5 Click **Login**.

The Main Menu page appears.

Step 6 Click the license link at the top of the Main Menu page, or choose **Configuration > License Keys**.

The Manage License Keys page appears.

Step 7 In the License File field, enter the complete path to the location of the license file on your local system. Remember to include the name of the license file while specifying the pathname. Or, click **Browse**.

The details regarding the license file appear. For details on licensing in this release, see the *Cisco Prime Cable Provisioning 6.1.3 User Guide*.

Accessing the DPE CLI

To access the DPE CLI, open a Telnet session to port 2323 from a local or remote host. Before you proceed, however, familiarize yourself with the access levels on the DPE.

Prime Cable Provisioning specifies a certain access level to authorize DPE access. Table 1-1 identifies the two access levels, which are also known as command modes. Each mode provides access to a specific set of commands.

Table 1-1 Command Modes on the DPE CLI

Mode	Description	Prompt
Login	Enables user commands for viewing the system configuration which requires PRIV_DPE_READ. In addition, to view device configuration PRIV_DEVICE_READ is required.	`bac_dpe>`
Enable	Enables privileged user commands for viewing, setting, and changing the system configuration, state, and data. Enable mode is controlled by PRIV_DPE_UPDATE and PRIV_DPE_SECURITY privileges.	`bac_dpe#`

Use the enable, page 2-5, and disable, page 2-4, commands to switch between the two modes.

You can access the DPE CLI following the steps described in:

- Accessing the DPE CLI from a Local Host, page 1-3
- Accessing the DPE CLI from a Remote Host, page 1-4

DPE CLI Privileges

Privileges required to access DPE CLI are:

Table 1-2 DPE CLI Privileges

Privilege	Description
PRIV_DPE_READ	Allows you to enter into login mode and view DPE status and settings.
PRIV_DPE_UPDATE	Allows you to enter into enable mode and set DPE properties and controlling of DPE lifecycle.
PRIV_DPE_SECURITY	Enables all security related Admin operations including changing DPE admin password, configuring authentication and shared secrets.
PRIV_DEVICE_READ	Enables viewing of device properties, searching for devices, and selecting devices. Also permits use of show device-config command in the DPE CLI.

For the complete list of default privileges in Prime Cable Provisioning see the Default Privileges section of the *Cisco Prime Cable Provisioning 6.1.3 User Guide*.

Accessing the DPE CLI from a Local Host

To access the CLI from a local host, you can use:

```
# telnet local_hostname 2323
```

where *local_hostname* specifies the name of the local host.

Or, you can use:

```
# telnet 0 2323
```

Defaults

Once you access the CLI, enter the DPE username and password to continue. The default login username is **admin** and password is **changeme**. Unlike the earlier releases of Prime Cable Provisioning, there is no need for second challenge (entering of password) to enter into enable mode. User can enter into enable mode based on the assigned privileges. For the list of DPE CLI privileges, see DPE CLI Privileges, page 1-3.

Note Although the default DPE username is admin and password is **changeme**, it is not the same as the one that you use to access the Prime Cable Provisioning Admin UI. The default admin user in DPE and RDU are two different users.

For information on how to change the login password, see password, page 2-7.

Examples

This result occurs when you access the DPE from a local host specifying its hostname.

```
bac_host# telnet local_bac_dpe 2323
Trying 10.10.2.25...
Connected to local_bac_dpe.example.com.
Escape character is '^]'.
```

```
Cisco Prime Cable Provisioning 5.1 (SOL_BAC5_1_0_00000000_0000)
Device Provisioning Engine local_bac_dpe

User Access Verification
Username: admin
Password: <changeme>

local_bac_dpe> enable
local_bac_dpe#
```

This result occurs when you access the DPE from a local host without specifying its hostname.

```
bac_host# telnet 0 2323
Trying 0.0.0.0...
Connected to 0.
Escape character is '^]'.

Cisco Prime Cable Provisioning 5.1 (SOL_BAC5_1_0_00000000_0000)
Device Provisioning Engine local_bac_dpe

User Access Verification
Username: admin
Password: <changeme>

bac_dpe> enable
bac_dpe#
```

Accessing the DPE CLI from a Remote Host

To access the CLI from a remote host, enter:

```
# telnet remote_hostname 2323
```

where *remote_hostname* specifies the name of the remote host.

Note If you cannot establish a Telnet connection to the CLI, the CLI server is probably not running. You may need to start the server. To start the server, enter:

```
# /etc/init.d/bprAgent start cli
```

Defaults

Once you access the CLI, you must enter the DPE username and password to continue. The default login username is **admin** and password is **changeme**.

Note Although the default DPE username is admin and password is changeme, it is not the same as the one that you use to access the Prime Cable Provisioning Admin UI. The default admin user in DPE and RDU are two different users.

For information on how to change the login password, see password, page 2-7.

Examples

This result occurs when you access the DPE from a remote host specifying its hostname.

```
bac_host# telnet remote_bac_dpe 2323
Trying 10.10.2.10...
```

```
Connected to remote_bac_dpe.example.com.
Escape character is '^]'.

Cisco Prime Cable Provisioning 5.1 (SOL_BAC5_1_0_00000000_0000)
Device Provisioning Engine remote_bac_dpe

User Access Verification
Username: admin
Password: <changeme>

remote_bac_dpe> enable
remote_bac_dpe#
```

Authentication Support

DPE CLI supports RADIUS and TACACS+ protocols for authenticating a user. Also the local user **admin** can be used to log into DPE CLI. You cannot configure both RADIUS and TACACS+ protocols together. Also, even when none of the protocols is configured, the local user **admin** can still be used for authentication. See Chapter 2, "System Commands" for details about the DPE CLI commands.

Local Authentication

This mode authenticates the default **admin** user in the local DPE and this mode is always enabled. In DPE CLI there is only one local account, admin. Users accessing the RDU cannot log into DPE CLI.

RADIUS Authentication

RADIUS is a UDP-based protocol used for enabling centralized authentication, authorization, and accounting for network access. It authenticates the users accessing the network services via the RADIUS server using the RADIUS standard protocol.

Cisco AV-pair needs to be configured in the RADIUS server to support authorization for DPE CLI RADIUS users. Cisco IOS/PIX 6.x is the RADIUS server that supports Cisco AV-pair in the Access Control Server (ACS) server. The Cisco AV-pair attribute value is:

cp:groups=<group-name>

For example:

cp:groups=Administrators

Here, Administrators is either the actual user group or user group mapping defined in the RDU. For more details, see the RADIUS Authentication section of the *Cisco Prime Cable Provisioning 6.1.3 User Guide*.

Note Any changes made to the user groups associated with the user will be reflected only in the next telnet session.

To enable backward compatibility, support of shell privileges priv-lvl=1 and priv-lvl=15 is continued.

Where, priv-lvl=1 is mapped to the privilege PRIV_DPE_READ and priv-lvl=15 is mapped to privileges PRIV_DPE_READ, PRIV_DEVICE_READ, PRIV_DPE_SECURITY, and PRIV_DPE_UPDATE.

Note Use of shell privileges is not a recommended method for authorizing DPE CLI RADIUS users. This method must be used only for backward compatibility.

TACACS+ Authentication

TACACS+ is a TCP-based protocol that supports centralized access control for several network devices and user authentication for the DPE CLI. Using TACACS+, a DPE supports multiple users (and their individual usernames) and the login password configured at the TACACS+ server. Here is how mapping of privileges is done in case of a TACACS+ server:

- The user must have priv-lvl=1 configured in the TACACS+ server for successful authentication. The user needs to have priv-lvl=15 configured in the TACACS+ server for entering into the enable mode.
- On successful authentication, user with priv-lvl=1 is assigned with the privilege PRIV_DPE_READ.
- On successful authorization in the enable mode, user with priv-lvl=15 is assigned with privileges PRIV_DPE_READ, PRIV_DEVICE_READ, PRIV_DPE_SECURITY, and PRIV_DPE_UPDATE.
- In the earlier release of Prime Cable Provisioning, user's password was required during login authentication and during the enable mode authentication. Now, password is not required during the enable mode authentication. Instead the password which has been entered during initial authentication is used for entering into the enable mode. Hence, the user should be configured with the same password for login authentication and enable mode authentication in the TACACS+ server.
- While logging into DPE CLI, if you enter the username as **admin**, the CLI falls back to local authentication mode. In this mode, you must enter both username and password. Once DPE CLI enters into local authentication mode, if wrong credentials are provided, the CLI prompts for the credentials again, now if TACACS+ username is entered, log in will not work. To log in using a TACACS+ username, the telnet session must be initiated again.
- If TACACS+ authentication is enabled in DPE CLI, but the server is not reachable, the CLI falls back to local authentication mode.
- When TACACS+ authentication is enabled in DPE CLI, if you enter wrong credentials accidentally, CLI prompts for username and password again. However, if you enter the username as **admin**, the CLI falls back to local authentication mode.

System Commands

This chapter describes the command-line interface (CLI) commands that you can use to manage and monitor the Prime Cable Provisioning Device Provisioning Engine (DPE).

If you run these 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 a DPE license.
```

The commands described in this chapter are:

Command	Description	CLI Mode		Required Privileges			
		Login	Enable	PRIV_DPE_READ	PRIV_DPE_UPDATE	PRIV_DPE_SECURITY	PRIV_DEVICE_READ
aaa authentication	Configures user authentication, authorization, and accounting services.		✓	✓	✓	✓	
disable	Exits the enable mode.		✓	✓	✓		
enable	Accesses the enable mode.	✓		✓	✓		
exit	Closes a Telnet connection to the DPE.	✓	✓	✓			
help	Displays a usage screen that assists you in using the commands on the CLI.	✓	✓	✓			
password	Changes the local system password, using which you can access the DPE.		✓	✓	✓	✓	
show clock	Displays the current system time and date.	✓	✓	✓			

Command	Description	CLI Mode		Required Privileges			
		Login	Enable	PRIV_DPE_READ	PRIV_DPE_UPDATE	PRIV_DPE_SECURITY	PRIV_DEVICE_READ
show commands	Displays all available commands on the CLI.	✓	✓	✓			
show disk	Identifies the disk that the DPE is currently using.	✓	✓	✓			
show hostname	Displays the hostname of the DPE.	✓	✓	✓			
show ip	Displays the current general IP settings configured on the DPE.	✓	✓	✓			
show ip route	Displays the IP routing table of the DPE.	✓	✓	✓			
show memory	Displays the current memory and swap space that are available on the DPE server.	✓	✓	✓			
show running-config	Displays the current configuration on the DPE.	✓	✓	✓			
show tftp files	Displays the files that are stored in the DPE cache.		✓	✓	✓		
show version	Displays the current version of DPE software.	✓	✓	✓			
tacacs-server host	Adds the TACACS+ server host address to the list of hosts.		✓	✓	✓	✓	
no tacacs-server host	Removes the TACACS+ server host address from the list of hosts.		✓	✓	✓	✓	
tacacs-server retries	The maximum number of times the TACACS+ client tries to connect with the TACACS+ server.		✓	✓	✓	✓	

Command	Description	CLI Mode		Required Privileges			
		Login	Enable	PRIV_DPE_READ	PRIV_DPE_UPDATE	PRIV_DPE_SECURITY	PRIV_DEVICE_READ
tacacs-server timeout	Sets the maximum length of time that the TACACS+ client waits for a response from the TACACS+ server.		✓	✓	✓	✓	
radius-server host	Adds the RADIUS server host address to the list of hosts.		✓	✓	✓	✓	
no radius-server host	Removes the RADIUS server host address from the list of hosts.		✓	✓	✓	✓	
radius-server retries	The maximum number of times the RADIUS client tries to connect with the RADIUS server.		✓	✓	✓	✓	
radius-server timeout	Sets the maximum length of time that the RADIUS client waits for a response from the RADIUS server.		✓	✓	✓	✓	
uptime	Shows the time during which the system is operational.	✓	✓	✓			

aaa authentication

Use the **aaa authentication** command to configure the CLI for user authentication, authorization, and accounting services using the local login or remote TACACS+ or RADIUS servers. This setting applies to all Telnet and console CLI interfaces.

Syntax Description

aaa authentication {**tacacs** | **radius**}

- **tacacs**—In this mode, the CLI server sequentially attempts a TACACS+ exchange with each server in the TACACS+ server list. The attempts continue for a specified number of retries. If the CLI reaches the end of the server list without a successful protocol exchange, a message is displayed indicating that the servers were not reachable. The CLI again prompts for the username and password. Enter the local CLI admin username and password to gain access to the CLI even if the TACACS+ service is unavailable.

- **radius**—In this mode, user authentication is performed via RADIUS server. The RADIUS server authentication details are similar to TACACS+ server. Cisco AV-pair needs to be configured in the RADIUS server to support DPE CLI RADIUS authentication. Cisco IOS/PIX 6.x is the RADIUS server that supports Cisco AV-pair in the Access Control Server (ACS) server. The Cisco AV-pair attribute value is:

  cp:groups=<group-name>

For example:

cp:groups=Administrators

Note When you telnet to DPE CLI, you are prompted to enter the username and password. You can either enter the username and password of the local DPE CLI admin user or a user configured in TACACS or Radius. At any given time, either of the TACACS or Radius server is enabled.

Defaults

AAA authentication is always enabled for the local admin user, even when RADIUS or TACACS+ is not configured.

Examples

This result occurs when you enable user authentication in the TACACS+ mode.

```
bac_dpe# aaa authentication tacacs
% OK
```

This result occurs when you enable user authentication in the radius mode.

```
bac_dpe# aaa authentication radius
% OK
```

disable

Use the **disable** command to exit the enable mode on the DPE. Once you exit the enable mode, you can view only those commands that relate to system configuration.

Syntax Description

No keywords or arguments.

Defaults

No default behavior or values.

Examples

```
bac_dpe# disable
bac_dpe>
```

enable

Use the **enable** command to access the DPE in the enable mode. You need not access the enable mode to view the system configuration; however, only in this mode can you change the system configuration, state, and data.

You must have the PRIV_DPE_UPDATE privilege to enter the enable mode using **enable** command.

Syntax Description No keywords or arguments.

Defaults The default password to access the enable mode is **changeme**.

Examples

```
bac_dpe> enable
bac_dpe#
```

This result occurs if you do not have the PRIV_DPE_UPDATE privilege.

```
bac_dpe# enable
Sorry, insufficient privileges.
```

exit

Use the **exit** command to close a Telnet connection to the DPE and return to the login prompt. After running this command, a message indicates that the Telnet connection has been closed.

Syntax Description No keywords or arguments.

Defaults No default behavior or values.

Examples This result occurs when you have accessed the CLI by specifying the hostname of the DPE.

```
bac_dpe# exit
% Connection closed.
Connection to 10.10.2.10 closed by foreign host.
```

This result occurs when you have accessed the CLI without specifying the hostname.

```
bac_dpe# exit
% Connection closed.
Connection to 0 closed by foreign host.
```

This result occurs when the Telnet connection closes because the CLI has been idle and the timeout period expired.

```
bac_dpe#
```

```
% Connection timed out.
Connection to 0 closed by foreign host.
```

help

Use the **help** command to display a help screen that can assist you in using the DPE CLI. If you need help on a particular command, or to list all available commands, enter *command* **?** or **?**, respectively.

Once you enter the command, a screen prompt appears to explain how you can use the help function.

Command Types

Two types of help are available:

1. Full help is available when you are ready to enter a command argument, such as **show ?**, and describes each possible argument.
2. Partial help is available when you enter an abbreviated argument and want to know what arguments match the input; for example, **show c?**.

Syntax Description

No keywords or arguments.

Defaults

No default behavior or values.

Examples

This result occurs when you use the **help** command.

```
bac_dpe# help
Help may be requested at any point in a command by entering a question mark '?'. If
nothing matches, the help list will be empty and you must backup until entering a '?'
shows the available options.

1) Full help is available when you are ready to enter a command argument (e.g. 'show ?')
and describes each possible argument.

2) Partial help is provided when an abbreviated argument is entered and you want to know
what arguments match the input (e.g. "show c?").
```

This result occurs when you invoke the full help function for a command; for example, **show ?**.

Note The **help** command output differs depending on the mode–login or enable–in which you run the command.

```
bac_dpe# show ?

  bundles         Shows the archived bundles.
  clock           Shows the current system time.
  commands        Shows the full command hierarchy.
  device-config   Show device configuration
  disk            Shows the current disk usage.
  dpe             Shows the status of the DPE process if started.
  hostname        Shows the system hostname.
  ip              Shows IP configuration details.
```

```
log             Shows recent log entries.
memory          Shows the current memory usage.
running-config  Shows the DPE configuration.
tftp            Shows TFTP details.
version         Shows DPE version.
```

This result occurs when you invoke the partial help function for arguments of a command; for example, **show clock**.

```
bac_dpe# show c?
clock     commands  cpu
bac_dpe# show clock
Thu Oct 25 01:20:14 EDT 2007
```

password

Use the **password** command to change the local system password, which you use to access the DPE. The system password is changed automatically for future logins and for FTP access.

Note The changes that you introduce through this command take effect for new users, but users who are currently logged in are not disconnected.

Syntax Description

password *password*

password—Identifies the new DPE password.

Defaults

The default password for accessing the DPE is **changeme**.

Examples

This result occurs when you change the password without being prompted (using an approach easier for scripting).

```
bac_dpe# password password2
Password changed successfully.
```

This result occurs when you are prompted for the password, and the password is changed successfully.

```
bac_dpe# password
New password: <password1>
Retype new password: <password1>
Password changed successfully.
```

This result occurs when you enter an incorrect password.

```
bac_dpe# password
New password: <password1>
Retype new password: <paswsord1>
Sorry, passwords do not match.
```

show

Use the **show** command to view system settings and status. Table 2-1 lists the keywords that you can use with this command.

Note To view the output for **show disk, show ip, show ip route**, and **show memory** on Linux, see *man mpstat*.

Table 2-1 List of show Commands

Command	Description	
show clock	Displays the current system time and date.	
	Syntax Description No keywords or arguments.	**Defaults** No default behavior or values.

	Example
	This result occurs when you run the show clock command:
	`bac_dpe# show clock`
	`Thu Oct 25 01:20:14 EDT 2007`	

Table 2-1 List of show Commands (continued)

Command	Description
show commands	Displays all commands on the DPE depending on the mode (login or enable) in which you access the CLI.

Syntax Description

No keywords or arguments.

Defaults

No default behavior or values.

Examples

This result occurs in the login mode.

```
bac_dpe> show commands
> enable
> exit
> help
> show bundles
> show clock
> show commands
> show device-config duid <DUID>
> show device-config mac <mac-address>
> show disk
> show dpe
> show dpe config
> show hostname
> show ip
> show ip route
> show log
> show log last <1..9999>
> show log run
> show memory
> show running-config
> show version
> uptime
```

Note The output presented in these examples is trimmed.

This result occurs in the enable mode.

```
bac_dpe# show commands
> aaa authentication radius
> aaa authentication tacacs
> clear bundles
> clear cache
> clear logs
> debug dpe cache
> debug dpe connection
> debug dpe dpe-server
> debug dpe event-manager
> debug dpe exceptions
> debug dpe framework
> debug dpe messaging
> debug on
> debug service packetcable 1 netsnmp
> debug service packetcable 1 registration
> debug service packetcable 1 registration-detail
> debug service packetcable 1 snmp
> debug service tftp 1 <ipv4|ipv6>
> disable
> [more]
```

To view the commands that flow beyond your screen, place the cursor at the `[more]` prompt and press **Spacebar**.

Table 2-1 List of show Commands (continued)

Command	Description	
show disk	Identifies the disk that the DPE is currently using. Once you enter the command, disk drive statistics appear.	
	Syntax Description	**Defaults**
	No keywords or arguments.	No default behavior or values.
show hostname	Displays the hostname configured for the DPE.	
	Syntax Description	**Defaults**
	No keywords or arguments.	No default behavior or values.

	Example
	`bac_dpe# show hostname`
	`hostname = bac_dpe.example.com`	

show ip	Displays the current general IP settings configured on the DPE. The DPE uses these settings when it reboots. For specific interface settings, use the **show interface** commands.	
	Syntax Description	**Defaults**
	No keywords or arguments.	No default behavior or values.
show ip route	Displays the IP routing table of the DPE, including any custom routes. The default gateway is indicated by the G flag in the flags column.	
	Syntax Description	**Defaults**
	No keywords or arguments.	No default behavior or values.
show memory	Displays the current memory and swap space that are available on the device running the DPE.	
	Syntax Description	**Defaults**
	No keywords or arguments.	No default behavior or values.

Table 2-1 List of show Commands (continued)

<table>
<tr><th>Command</th><th>Description</th></tr>
<tr><td>show running-config</td><td>Displays the current configuration on the DPE.
Syntax Description
No keywords or arguments.
Defaults
No default behavior or values.
Example
<pre>bac_dpe# show running-config
     dpe port 49186
     dpe provisioning-group primary default
     dpe rdu-server bacdev2-t5220-1-d8 49187
     dpe shared-secret <value is set>
     log level 5-notification
     no debug all
     no debug dpe cache
     no debug dpe connection
     no debug dpe device-config-compression
     no debug dpe device-config-compression-details
     no debug dpe device-config-decompression
     no debug dpe device-config-decompression-details
     no debug dpe dpe-server
     no debug dpe event-manager
     no debug dpe exceptions
     no debug dpe framework
     no debug dpe messaging
     no debug service packetcable 1 netsnmp
     no debug service packetcable 1 registration
     no debug service packetcable 1 registration-detail
     no debug service packetcable 1 snmp
     no dpe docsis emic-shared-secret
     no dpe docsis shared-secret
     no dpe provisioning-group secondary
     no service packetcable 1 snmp key-material
     radius-server retries 3
     radius-server timeout 3
     service tftp 1 ipv4 verify-ip
     service tftp 1 ipv6 verify-ip
     snmp-server community baccread ro
     snmp-server community baccwrite rw
     snmp-server contact <unknown>
     snmp-server location <unknown>
     snmp-server udp-port 8001
     tacacs-server retries 2
     tacacs-server timeout 5</pre></td></tr>
</table>

Table 2-1 List of show Commands (continued)

<table>
<tr><th>Command</th><th>Description</th></tr>
<tr><td>show tftp files</td><td>Displays the files that are stored in the DPE cache.
You cannot use this command to display the files that are stored in the local directory.
Syntax Description | Defaults
No keywords or arguments. | The default is 500.
Example
This result occurs when you run the show tftp files command:
<pre>bac_dpe# show tftp files

The list of TFTP files currently in DPE cache

filename   size

bronze.cm   310
gold.cm   310
silver.cm   310
unprov.cm   310
unprov_11.cm   320
unprov_30.cm   264
unprov_30v4.cm   152
unprov_30v6.cm   196
unprov_packet_cable.bin   333
unprov_wan_man.cfg   72

DPE caching 10 external files.
Listing the first 10 files, 0 files omitted</pre></td></tr>
<tr><td>show version</td><td>Displays the current version of DPE software.
Syntax Description | Defaults
No keywords or arguments. | No default behavior or values.
Example
This result occurs when you run the show version command:
<pre>bac_dpe# show version
Version: BAC 5.1 (BAC_LNX_TRUNK_20121203_2231_1128)</pre></td></tr>
</table>

tacacs-server

Use the **tacacs-server** command to configure user authentication settings in TACACS+. Table 2-2 lists the keywords that you can use with this command.

Table 2-2 List of tacacs-server Commands

Command	Description
tacacs-server host	Adds the TACACS+ server host address to the list of hosts. When you enable TACACS+ authentication, the client attempts to authenticate the user with the first reachable server. If the authentication succeeds the user is allowed to log in depending on the privileges obtained from the user group specified in the CISCO AV Pair (cp:groups). If the first server is not reachable, then the next server in the list is attempted till the list exhausts. To remove a TACACS+ server from the list of TACACS+ servers in the CLI, use the **no** form of this command. See no tacacs-server host, page 2-14.
	Syntax Description **tacacs-server host** *host* [**key** *encryption-key*] • *host*—Specifies the IP address or the hostname of the TACACS+ server. • *encryption-key*—Identifies the encryption key (optional).
	Defaults No default behavior or values.

	Examples
	
	This result occurs when you add a TACACS+ server using its IP address (10.0.1.1) without encryption.
	
	`bac_dpe# tacacs-server host 10.0.1.1`
	`% OK`
	
	This result occurs when you add a TACACS+ server using its IP address (10.0.1.1) and an encryption key (hg667YHHj).
	
	`bac_dpe# tacacs-server host 10.0.1.1 key hg667YHHj`
	`% OK`
	
	This result occurs when you add a TACACS+ server using its hostname (tacacs1.cisco.com) without encryption.
	
	`bac_dpe# tacacs-server host tacacs1.example.com`
	`% OK`
	
	This result occurs when you add a TACACS+ server using its hostname (tacacs1.cisco.com) and an encryption key (hg667YHHj).
	
	`bac_dpe# tacacs-server host tacacs1.example.com key hg667YHHj`
	`% OK`

Table 2-2 List of tacacs-server Commands (continued)

Command	Description
no tacacs-server host	Removes the TACACS+ server host address from the list of hosts. To add a TACACS+ server, see tacacs-server host, page 2-13.
	Syntax Description **no tacacs-server host** *host* *host*—Specifies either the IP address or the hostname of the TACACS+ server. **Defaults** No default behavior or values.

	Examples
	This result occurs when you remove a TACACS+ server using its IP address.
	`bac_dpe# no tacacs-server host 10.0.1.1`
	`% OK`
	This result occurs when you remove a TACACS+ server using its hostname.
	`bac_dpe# no tacacs-server host tacacs1.example.com`
	`% OK`

tacacs-server retries	Sets the maximum number of times the TACACS+ protocol exchange is tried before the TACACS+ client considers a specific TACACS+ server unreachable. When this limit is reached, the TACACS+ client moves to the next server in its TACACS+ server list till the list has been exhausted.
	Syntax Description **tacacs-server retries** *value* *value*—Specifies a dimensionless number from 1 to 100. This value applies to all TACACS+ servers. **Defaults** The default is 3.

	Example
	This result occurs when you configure retry value for TACACS+ server:
	`bac_dpe# tacacs-server retries 10`
	`% OK`

tacacs-server timeout	Sets the maximum length of time that the TACACS+ client waits for a response from the TACACS+ server before it considers the protocol exchange to have failed.
	Syntax Description **tacacs-server timeout** *value* *value*—Specifies the maximum length of time that the TACACS+ client waits for a TACACS+ server response. This value must be from 1 to 300 seconds, and applies to all TACACS+ servers. **Defaults** The default is 5 seconds.

	Example
	This result occurs when you configure timeout value for TACACS+ server:
	`bac_dpe# tacacs-server timeout 10`
	`% OK`

radius-server

Use the **radius-server** command to configure user authentication settings in RADIUS. Table 2-3 lists the keywords that you can use with this command.

Table 2-3 List of radius-server Commands

Command	Description
radius-server host	Adds the RADIUS server host address to the list of hosts. When you enable RADIUS authentication, the client attempts to authenticate the user with the first reachable server. If the authentication succeeds, the user is allowed to login depending on the privileges obtained from the user group specified in the CISCO AV Pair (cp:groups). If the first server is not reachable then the next server in the list is attempted till the list exhausts. The order of the commands that appears in show run is the order in which they are contacted. To remove a RADIUS server from the list of RADIUS servers in the CLI, use the no form of this command. See no radius-server host, page 2-16.
	Syntax Description **radius-server host** *host* [**key** *encryption-key*] [**port** *port-number*] • *host*—Specifies the IP address or the hostname of the RADIUS server. • *encryption-key*—Identifies the encryption key (optional). • *port-number*—Identifies the port number (optional). **Defaults** No default behavior or values.

	Examples
	
	This result occurs when you add a RADIUS server using its IP address with key and port number.
	
	`bac_dpe# radius-server host 10.10.10.10 key secret port 1812`
	`% OK`

Table 2-3 List of radius-server Commands (continued)

Command	Description	
no radius-server host	Removes the RADIUS server host address from the list of hosts. For details about adding a RADIUS server, see radius-server host, page 2-15.	
	Syntax Description **no radius-server host** *host* *host*—Specifies either the IP address or the hostname of the RADIUS server.	**Defaults** No default behavior or values.

	Examples
	This result occurs when you remove a RADIUS server using its IP address:
	`bac_dpe# no radius-server host 10.10.10.10`
	`% OK`	

radius-server retries	Sets the maximum number of times the RADIUS protocol exchange is tried before the RADIUS client considers a specific RADIUS server unreachable. When this limit is reached, the RADIUS client moves to the next server in its RADIUS server list till the list has been exhausted.	
	Syntax Description **radius-server retries** *value* *value*—Specifies a dimensionless number from 1 to 10. This value applies to all RADIUS servers.	**Defaults** The default is 3.

	Example
	This result occurs when you configure retry value for RADIUS server:
	`bac_dpe# radius-server retries 10`
	`% OK`	

radius-server timeout	Sets the maximum length of time that the RADIUS client waits for a response from the RADIUS server before it considers the protocol exchange to have failed.	
	Syntax Description **radius-server timeout** *value* *value*—Specifies maximum length of time that the RADIUS client waits for a RADIUS server response. This value must be from 1 to 30 seconds, and applies to all RADIUS servers.	**Defaults** The default is 3 seconds.

	Example
	This result occurs when you configure timeout value for RADIUS server:
	`bac_dpe# radius-server timeout 5`
	`% OK`	

uptime

Use the **uptime** command to identify how long the system has been operational. This information is useful for determining how frequently the device is rebooted. It is also helpful when checking the reliability of the DPE when it is in a stable condition.

Syntax Description

No keywords or arguments.

Defaults

No default behavior or values.

Examples

```
bac_dpe# uptime
  1:47am  up 496 day(s),  8:49,  1 user,  load average: 0.14, 0.07, 0.06
```

DPE Configuration Commands

This chapter describes the command-line interface (CLI) commands that you can use to manage and monitor the Prime Cable Provisioning Device Provisioning Engine (DPE).

The commands described in this chapter are:

Command	Description	CLI Mode		Required Privileges			
		Login	Enable	PRIV_DPE_READ	PRIV_DPE_UPDATE	PRIV_DPE_SECURITY	PRIV_DEVICE_READ
clear cache	Erases the DPE cache and resets the server to a clean state.		✓	✓	✓		
dpe docsis shared-secret	Sets a DOCSIS shared secret on the DPE.		✓	✓	✓	✓	
dpe docsis emic-shared-secret	Sets a DOCSIS EMIC shared secret on the DPE.		✓	✓	✓	✓	
dpe port	Sets the port number that the DPE uses to communicate with Cisco Network Registrar extensions.		✓	✓	✓		
dpe provisioning-group primary	Sets the DPE in a specific primary provisioning group.		✓	✓	✓		
dpe provisioning-group secondary	Sets secondary provisioning groups for the DPE.		✓	✓	✓		
dpe rdu-server port	Specifies the port to connect to the RDU.		✓	✓	✓		

Command	Description	CLI Mode		Required Privileges			
		Login	Enable	PRIV_DPE_READ	PRIV_DPE_UPDATE	PRIV_DPE_SECURITY	PRIV_DEVICE_READ
dpe rdu-server source ip	Configures the DPE source interface to connect to the RDU.		✓	✓	✓		
dpe rdu-server source port	Configures the DPE source port to connect to the RDU.		✓	✓	✓		
dpe reload	Restarts the DPE.		✓	✓	✓		
dpe shared-secret	Sets the shared secret used in communications with the RDU.		✓	✓	✓	✓	
dpe start | stop	Starts or stops the DPE.		✓	✓	✓		
dpe truststore-password	Sets the truststore password.		✓	✓	✓	✓	
interface ip pg-communication	Configures an interface to communicate with Cisco Network Registrar extensions.		✓	✓	✓		
interface ip provisioning	Configures an interface to handle provisioning requests.		✓	✓	✓		
interface ip provisioning fqdn	Sets the fully qualified domain name for a specific interface.		✓	✓	✓		
service tftp allow-read-access	Enables TFTP read requests from the file system.		✓	✓	✓		
service tftp ipv4 | ipv6 blocksize	Enables or disables the blocksize option for the TFTP service for IPv4 or IPv6.		✓	✓	✓		
service tftp ipv4 | ipv6 enabled	Enables or disables the TFTP service for IPv4 or IPv6.		✓	✓	✓		
service tftp ipv4 | ipv6 verify-ip	Enables the verification of requestor IP addresses on dynamic configuration TFTP requests.		✓	✓	✓		
service tod	Enables or disables the ToD service for IPv4 or IPv6.		✓	✓	✓		

Command	Description	CLI Mode		Required Privileges			
		Login	Enable	PRIV_DPE_READ	PRIV_DPE_UPDATE	PRIV_DPE_SECURITY	PRIV_DEVICE_READ
show device-attribute	Displays the last transaction time.		✓	✓			
dump device-attributes	Dumps all the device attributes from a DPE.		✓		✓		
show dump-device-attributes-status	Displays the status of the dumping process.		✓	✓			
show device-config	Displays a device configuration that is cached at the DPE.		✓	✓			✓
show dpe	Displays the state of the DPE process and, if running, its operational statistics.	✓	✓	✓			
show dpe config	Displays the current settings on the DPE.	✓	✓	✓			

clear cache

Use the **clear cache** command to erase the DPE cache and reset the server to a clean state. When the DPE is restarted, it connects to the RDU and rebuilds the cache from the information stored in the RDU database.

Note Before erasing the DPE cache, ensure that you stop the DPE by running the **dpe stop** command. For more information, see dpe start | stop, page 3-13.

You should clear the cache only when the DPE encounters a major problem. Running this command forces the DPE to rebuild or repopulate its device cache. This process may take an extended period of time to complete.

Once the command is entered, the DPE cache is cleared and a prompt appears to indicate the amount of disk space cleared as a result. If the cache could not be cleared, the reason for the failure appears.

Syntax Description

No keywords or arguments.

Defaults

No default behavior or values.

Examples This result occurs when the cache is successfully cleared.

```
bac_dpe# clear cache
Clearing DPE cache...
+ 820224 bytes cleared.
```

This result occurs when the cache has already been cleared.

```
bac_dpe# clear cache
Clearing DPE cache...
+ Cache already cleared.
```

This result occurs when the DPE has not been stopped.

```
bac_dpe# clear cache
DPE must be stopped before clearing cache.
```

dpe docsis shared-secret

Use the **dpe docsis shared-secret** command to set a DOCSIS shared secret (DSS) on the DPE. The DSS is used to calculate the message integrity check of cable modems and the cable modem termination system (CMTS).

Note While setting or changing the DSS, we recommend that you use a secure connection.

To disable the DSS, use the **no** form of this command.

Syntax Description **dpe docsis shared-secret** *type secret*

- *type*—Identifies whether the shared secret string appears as clear text or as encrypted text.

  To specify the format, enter:

  - 0 for a clear text string. This string is the default setting.
  - 7 for a Cisco IOS-encrypted shared-secret text string.
- *secret*—Identifies the secret string. You must enter a value from 2 to 132 characters.

If, after running this command, you use the **show running-config** command, a new line appears identifying the shared secret and its type.

Defaults The DSS is by default not configured on the DPE.

Examples

```
bac_dpe# dpe docsis shared-secret 0 changeme
% OK (Warning: Current input accepted. Note a secure connection is recommended to set or
change the DOCSIS Shared Secret.)
```

no dpe docsis shared-secret

Use the **no dpe docsis shared-secret** command to disable the DOCSIS shared secret (DSS) configured on the DPE.

To enable the DSS, see dpe docsis shared-secret, page 3-4.

Syntax Description No keywords or arguments.

Defaults The DSS is by default not configured on the DPE.

Examples

```
bac_dpe# no dpe docsis shared-secret
% OK
```

dpe docsis emic-shared-secret

Use the **dpe docsis emic-shared-secret** command to set a Secondary DOCSIS Shared Secret (SDSS) on the DPE. The SDSS is used to calculate the message integrity check of cable modems and the Cable Modem Termination System (CMTS) with DOCSIS 3.0.

Note We recommend that you use a secure connection while setting or changing the SDSS.

To disable the SDSS, use the **no** form of this command.

Syntax Description **dpe docsis emic-shared-secret** *type secret*

- *type*—Identifies whether the secondary shared secret string appears as clear text or as encrypted text.

  To specify the format, enter:

  - 0 for a clear text string. This string is the default setting.
  - 7 for a shared secret in PBKDF2_DES3CBC encrypted form.
- *secret*—Identifies the secret string. You must enter a value that has 2 to 200 characters.

If, after running this command, you run the **show running-config** command, a new line appears identifying the shared secret and its type.

Defaults By default, the SDSS is not configured on the DPE.

Examples

```
bac_dpe# dpe docsis emic-shared-secret 0 changeme
% OK (Warning: Current input accepted. Note a secure connection is recommended to set or
change the secondary DOCSIS Shared Secret.)
```

no dpe docsis emic-shared-secret

Use the **no dpe docsis emic-shared-secret** command to disable the SDSS configured on the DPE. A DPE reload is required after executing this command. See dpe reload, page 3-12

For details about enabling the SDSS, see dpe docsis emic-shared-secret.

Syntax Description

No keywords or arguments.

Defaults

By default, the SDSS is not configured on the DPE.

Examples

```
bac_dpe# no dpe docsis emic-shared-secret
% OK (Requires DPE restart "> dpe reload")
```

dpe port

Use the **dpe port** command to specify the port number that the DPE uses to communicate with the Network Registrar extension points. You can leave this port number intact unless there is a need to change it for firewall reasons.

Note You must stop the DPE before changing the port number. If you attempt to run this command on an operational DPE, the following error message appears:

```
ERROR: DPE must be stopped before changing the port number.
```

The changes that you introduce through this command do not take effect until you restart the DPE. For information on stopping and starting the DPE, see dpe start | stop, page 3-13.

Syntax Description

dpe port *port*

port—Identifies the port number assigned for connecting to the DPE.

Defaults

The default port that the DPE uses is 49186.

Examples

```
bac_dpe# dpe port 49186
% OK
```

dpe provisioning-group primary

Use the **dpe provisioning-group primary** command to specify the DPE as a member of a specified primary provisioning group. Most DPEs are configured with a primary provisioning group; however, selecting multiple provisioning groups allows multiple DHCP servers to use this DPE.

Note If you enable PacketCable voice technology, ensure that a DPE belongs to only one provisioning group.

When assigning new provisioning groups that have a large number of devices, restarting the DPE can take an extended period of time depending on the number of devices in your network and the size of the device configurations. This delay occurs because the cache for each provisioning group has to be synchronized or, for new provisioning groups, completely rebuilt.

Note Typically, you must change the provisioning groups only when the DPE is first deployed on the network.

After you use this command, run the **dpe reload** command so that the changes take effect. See dpe reload, page 3-12.

To remove any configured primary provisioning groups, use the **no** form of this command. See no dpe provisioning-group primary, page 3-8.

Syntax Description

dpe provisioning-group primary *name* [*name**]

- *name*—Identifies the assigned primary provisioning group.
- *name**—Allows the entry of multiple provisioning groups. When specifying multiple provisioning groups, you must insert a space between their names.

Defaults

The default primary provisioning group is the provisioning group that you configure as the default.

You can use any name to identify the primary provisioning group. By default, however, the primary provisioning group is identified as ‘default’.

Examples

This result occurs when you specify a single primary provisioning group.

```
bac_dpe# dpe provisioning-group primary PrimaryProvGroup
% OK (Requires DPE restart "> dpe reload")
```

This result occurs when you specify multiple primary provisioning groups.

```
bac_dpe# dpe provisioning-group primary provisioning-grp-1 provisioning-grp-2
% OK (Requires DPE restart "> dpe reload")
```

no dpe provisioning-group primary

Use the **no dpe provisioning-group primary** command to clear configured primary provisioning groups. If primary provisioning groups are not available, you can use the DPE as a backup for other provisioning groups or as a TFTP file cache.

Note Every DPE must belong to at least one primary or secondary provisioning group.

After you use this command, run the **dpe reload** command so that the changes take effect. See dpe reload, page 3-12.

To specify the DPE as a member of a specified primary provisioning group, see dpe provisioning-group primary, page 3-7.

Syntax Description No keywords or arguments.

Defaults No default behavior or values.

Examples

```
bac_dpe# no dpe provisioning-group primary
% OK (Requires DPE restart "> dpe reload")
```

dpe provisioning-group secondary

Use the **dpe provisioning-group secondary** command to set secondary provisioning groups for the DPE server to use. Most DPEs are configured with a primary provisioning group; however, selecting multiple provisioning groups allows multiple DHCP servers to use this DPE.

Note Secondary provisioning groups are used for provisioning only when the primary provisioning groups are not available or are overloaded.

When assigning new provisioning groups that have a large number of devices, restarting the DPE can take an extended period of time depending on the number of devices in your network and the size of the device configurations. This delay occurs because the cache for each provisioning group has to be synchronized or, for new provisioning groups, completely rebuilt.

Note Typically, you must change the provisioning groups only when the DPE is first deployed on the network.

After you use this command, run the **dpe reload** command so that the changes take effect. See dpe reload, page 3-12.

To remove any configured secondary provisioning groups, use the **no** form of this command. See no dpe provisioning-group secondary, page 3-9.

Syntax Description **dpe provisioning-group secondary** *name* [*name**]

- *name*—Identifies the assigned secondary provisioning group.
- *name**—Allows the entry of multiple provisioning groups. When specifying multiple provisioning groups, you must insert a space between their names.

Defaults No default behavior or values.

Examples This result occurs when you specify a single secondary provisioning group.

```
bac_dpe# dpe provisioning-group secondary SecondaryProvGroup
% OK (Requires DPE restart "> dpe reload")
```

This result occurs when you specify multiple secondary provisioning groups.

```
bac_dpe# dpe provisioning-group primary provisioning-second-1 provisioning-second-2
% OK (Requires DPE restart "> dpe reload")
```

no dpe provisioning-group secondary

Use the **no dpe provisioning-group secondary** command to clear configured secondary provisioning groups. If secondary provisioning groups are not available, the DPE can be used as a primary in other provisioning groups.

After you use this command, run the **dpe reload** command so that the changes take effect. See dpe reload, page 3-12.

To set secondary provisioning groups for the DPE, see dpe provisioning-group secondary, page 3-8.

Syntax Description No keywords or arguments.

Defaults No default behavior or values.

Examples

```
bac_dpe# no dpe provisioning-group secondary
% OK (Requires DPE restart "> dpe reload")
```

dpe rdu-server

Use the **dpe rdu-server** command to configure the DPE to connect to the RDU server. Table 3-1 lists the keywords that you can use with this command.

Table 3-1 List of dpe rdu-server Commands

Command	Description
dpe rdu-server port	Identifies the RDU to which the DPE connects. Normally, you configure the RDU on the default port, but for security reasons, you could configure it to run on a nondefault port. After you use this command, run the **dpe reload** command so that the changes take effect. See dpe reload, page 3-12.

Syntax Description

dpe rdu-server {*host* | *x.x.x.x*} *port secure*

- *host*—Identifies the fully qualified domain name (FQDN) of the RDU host.
- *x.x.x.x*—Identifies the IP address of the RDU host.
- *port*—Identifies the port number on which the RDU is listening for DPE connections.
- *secure*—Identifies whether to enable secure mode of communication with the RDU. The value can either be true or false where true indicates secure mode.

Defaults

The default port on which the RDU listens for the DPE is 49187.

Examples

This result occurs when you specify the RDU host:

- Using its FQDN.

```
bac_dpe# dpe rdu-server rdu.example.com 49187 false
% OK (Requires DPE and DPE CLI restart)
```

- Using its IP address.

```
bac_dpe# dpe rdu-server 10.10.20.1 49187 false
% OK (Requires DPE and DPE CLI restart)
```

- Enabling secure mode.

```
bac_dpe# dpe rdu-server 10.10.20.1 49188 true
% OK (Requires DPE and DPE CLI restart)
```

Table 3-1 List of dpe rdu-server Commands (continued)

<table>
<tr><th>Command</th><th>Description</th></tr>
<tr><td>dpe rdu-server source ip

no dpe rdu-server source ip</td><td>Configures the DPE to use the specified interface as its source when connecting to the RDU. If you do not specify an interface, the DPE allows the operating system to determine the interface to use while communicating with the RDU server.

Note While using this command, you can specify IP addresses only in the IPv4 format.

After you use this command, run the dpe reload command so that the changes take effect. See dpe reload, page 3-12.

To clear the configured interface, use the no form of this command. When clearing the configured interface, you need not specify the IP address of the interface.

Syntax Description

dpe rdu-server source ip ip_address [?]

• ip_address—Identifies the IP address of a specific DPE interface, in the IPv4 addressing format.
• ?—Dynamically determines and displays the available IP addresses. This parameter is optional. When you use this option, you need not specify an IP address.

Defaults

No default behavior or values.

Examples

This result occurs when you configure the DPE interface.

• Using its IP address

<pre>bac_dpe# dpe rdu-server source ip 10.10.1.2
% OK (Requires DPE restart "> dpe reload")</pre>• Without specifying its IP address

<pre>bac_dpe# dpe rdu-server source ip
% OK (Requires DPE restart "> dpe reload")</pre>• Using the ? option

<pre>bac_dpe# dpe rdu-server source ip ?
  <ip address> [10.10.1.2]
  <cr></pre>This result occurs when you clear the configured DPE interface.

<pre>bac_dpe# no dpe rdu-server source ip
% OK (Requires DPE restart "> dpe reload")</pre></td></tr>
</table>

Table 3-1 List of dpe rdu-server Commands (continued)

Command	Description
dpe rdu-server source port **no dpe rdu-server source port**	Configures the DPE to use the specified port as the source port when connecting to the RDU. If you do not specify the port, the DPE allows the operating system to determine the port to use while communicating with the RDU. After you use this command, run the **dpe reload** command so that the changes take effect. See dpe reload, page 3-12. To clear the configured port, use the **no** form of this command. When clearing the configured port, you need not specify the port number.
	Syntax Description **dpe rdu-server source port** *port* *port*—Identifies the number of the DPE source port. **Note** If the port you specify is not available, an error message appears. **Defaults** No default behavior or values.

	Examples
	This result occurs when you configure a port to communicate with the RDU.
	`bac_dpe# dpe rdu-server source port 49186`
	`% OK (Requires DPE restart "> dpe reload")`
	
	This result occurs when you clear the configured port through which the DPE communicates with the RDU.
	`bac_dpe# no dpe rdu-server source port`
	`% OK (Requires DPE restart "> dpe reload")`

dpe reload

Use the **reload** command to restart the DPE. It must be operational before you reload it. If the DPE does not stop within 60 seconds, the Prime Cable Provisioning process watchdog (bprAgent) forces the DPE to stop, and an alert message, indicating that the DPE has been stopped, appears. Once the message appears, the DPE restarts.

Syntax Description

No keywords or arguments.

Defaults

No default behavior or values.

Examples

```
bac_dpe# dpe reload
Process [dpe] has been restarted.
```

dpe shared-secret

Use the **dpe shared-secret** command to set the shared secret used for communications with the RDU. Communication fails if the shared secret, which is set on the two servers, is not the same.

After you use this command, run the **dpe reload** command so that the changes take effect. See dpe reload, page 3-12.

Syntax Description

dpe shared-secret *secret*

secret—Identifies the RDU shared secret.

Defaults

No default behavior or values.

Examples

```
bac_dpe# dpe shared-secret private
% OK (Requires DPE and DPE CLI restart)
```

dpe start | stop

Use the **dpe start | stop** command to start or stop the DPE.

Syntax Description

dpe start | stop

- **start**—Starts the DPE. You can use this command only when the DPE is not running. Having the DPE start successfully does not guarantee that the DPE will run successfully. Check the DPE log to ensure that the DPE has started correctly. Also, check the log periodically to ensure that no additional errors have occurred.
- **stop**—Stops the DPE. You can use this command only when the DPE is running. If the DPE has not stopped within 60 seconds, the DPE process watchdog (bprAgent) forces the DPE to stop, and an alert message, indicating that the DPE has been stopped, appears.

Defaults

No default behavior or values.

Examples This result occurs when the DPE is started.

```
bac_dpe# dpe start
Process [dpe] has been started
```

This result occurs if the DPE is started when it is already operational.

```
bac_dpe# dpe start
Process [dpe] is already running
```

This result occurs when the DPE is stopped.

```
bac_dpe# dpe stop
Process [dpe] has been stopped.
```

dpe truststore-password

Use the **dpe truststore-password** command to set the truststore (cacerts) password. By default, the password is set to changeit.

After you use this command, run the **dpe reload** command so that the changes take effect. See dpe reload, page 3-12.

Syntax Description **dpe truststore-password** *changeme*

changeme—Identifies the truststore password. You must enter a value from 8 to 20 characters.

Defaults No default behavior or values.

Examples

```
bac_dpe# dpe truststore-password changeme
% OK (Requires DPE and DPE CLI restart)
```

interface ip provisioning

Use the **interface ip provisioning** command to configure the specified interface, identified by its IP address, to handle provisioning requests. Only interfaces that have provisioning enabled are used for communication with devices and the DHCP server.

After you use this command, run the **dpe reload** command so that the changes take effect. See dpe reload, page 3-12.

To disable the configured interface, use the **no** form of this command. See no interface ip provisioning, page 3-15.

Syntax Description **interface ip** *ip_address* **provisioning** [**?**]

- *ip_address*—Specifies the IP address of the interface in the IPv4 or the IPv6 format.

- **?**—Dynamically determines and displays the available interfaces by their IP addresses. This parameter is optional. When you use this option, you need not specify an IP address.

The IP addresses that appear when you use the **?** option do not change after you install the CLI. If you want to change the provisioning IP address, manually remove the existing IP address and configure a new IP address in the following manner:

1. Delete the existing IP address, using the **no interface ip** *ip_address* **provisioning** command.
2. Shut down the CLI process, using the **/etc/init.d/bprAgent stop cli** command.
3. Change the IP address on the network card.
4. Start the CLI process again, using the **/etc/init.d/bprAgent start cli** command.
5. Add the new IP address from the DPE command line, using the **interface ip provisioning** command.
6. Reload the DPE, using the **dpe reload** command.

Defaults

No default behavior or values.

Examples

This result occurs when you configure an interface by specifying its IPv4 address.

```
bac_dpe# interface ip 10.10.10.133 provisioning
% OK (Requires DPE restart "> dpe reload")
```

This result occurs when you configure an interface by specifying its IPv6 address.

```
bac_dpe# interface ip 2001:0DB8:0:0:203:baff:fe12:d5ea provisioning
% OK (Requires DPE restart "> dpe reload")
```

This result occurs when you use the **?** option.

```
bac_dpe# interface ip ?
  10.10.10.133                       eri0 [3]
  2001:0DB8:0:0:203:baff:fe12:d5ea   eri0 [1]
  2001:0DB8:0:1:203:baff:fe12:d5ea   eri0
  fe80:0:0:0:203:baff:fe12:d5ea      eri0 [2]
```

no interface ip provisioning

Use the **no interface ip provisioning** command to disable provisioning via the specified interface.

After you use this command, run the **dpe reload** command so that the changes take effect. See dpe reload, page 3-12.

To enable an interface, see interface ip provisioning, page 3-14.

Syntax Description

no interface ip *ip_address* **provisioning** [**?**]

- *ip_address*—Specifies the IPv4 or IPv6 address of the interface.
- **?**—Dynamically determines and displays the available interfaces by their IP addresses. This parameter is optional. When you use this option, you need not specify an IP address.

Defaults No default behavior or values.

Examples This result occurs when you disable an interface by specifying its IPv4 address.

```
bac_dpe# no interface ip 10.10.10.133 provisioning
% OK (Requires DPE restart "> dpe reload")
```

This result occurs when you disable an interface by specifying its IPv6 address.

```
bac_dpe# no interface ip 2001:0DB8:0:0:203:baff:fe12:d5ea provisioning
% OK (Requires DPE restart "> dpe reload")
```

This result occurs when you use the **?** option.

```
bac_dpe# no interface ip ?
  10.10.10.133                        eri0 [3]
  2001:0DB8:0:0:203:baff:fe12:d5ea    eri0 [1]
  2001:0DB8:0:1:203:baff:fe12:d5ea    eri0
  fe80:0:0:0:203:baff:fe12:d5ea       eri0 [2]
```

interface ip provisioning fqdn

Use the **interface ip provisioning fqdn** command to set the FQDN for a specific interface. The provisioning FQDN is the domain name that is given to devices to contact the specific DPE interface.

Note Before setting the FQDN for an interface, ensure that provisioning is enabled on that interface. To enable provisioning on an interface, see interface ip provisioning, page 3-14.

After you use this command, run the **dpe reload** command so that the changes take effect. See dpe reload, page 3-12.

To clear the configured FQDN, use the **no** form of this command. See no interface ip provisioning fqdn, page 3-17.

Syntax Description **interface ip** *ip_address* **provisioning fqdn** *fqdn*

- *ip_address*—Identifies the interface on the DPE.
- *fqdn*—Identifies the FQDN that is set on the specified interface. This FQDN is sent as the SNMPEntity in DHCP option 177, suboption 3.

Defaults No default behavior or values.

Examples This result occurs when you set the FQDN of an IPv4 interface.

```
bac_dpe# interface ip 10.10.1.2 provisioning fqdn dpe.example.com
% OK (Requires DPE restart "> dpe reload")
```

This result occurs when you set the FQDN of an IPv6 interface.

```
bac_dpe# interface ip 2001:0DB8:0:0:203:baff:fe12:d5ea provisioning fqdn dpe.example.com
% OK (Requires DPE restart "> dpe reload")
```

no interface ip provisioning fqdn

Use the **no interface ip provisioning fqdn** command to clear the FQDN for a specific interface. The provisioning FQDN is the domain name that is given to devices to contact the specific DPE interface.

If you clear the last existing FQDN of an IPv4 interface when Packet Cable is enabled, the following error appears:

```
% Cannot remove this interface when PacketCable Service is enabled.
% Error processing command
```

After you run this command, run the **dpe reload** command so that the changes take effect. See dpe reload, page 3-12.

For details about setting the FQDN for an interface, see interface ip provisioning fqdn, page 3-16.

Syntax Description

no interface ip *ip_address* **provisioning fqdn** *fqdn*

- *ip_address*—Identifies the interface on the DPE.
- *fqdn*—Identifies the FQDN that is set on the specified interface. This FQDN is sent as the SNMPEntity in DHCP option 177, suboption 3.

Defaults

No default behavior or values.

Examples

This result occurs when you clear the FQDN of an interface by specifying its IPv4 address.

```
bac_dpe# no interface ip 10.10.1.2 provisioning fqdn dpe.example.com
% OK (Requires DPE restart "> dpe reload")
```

This result occurs when you clear the FQDN of an interface by specifying its IPv6 address.

```
bac_dpe# no interface ip 2001:0DB8:0:0:203:baff:fe12:d5ea provisioning fqdn
dpe.example.com
% OK (Requires DPE restart "> dpe reload")
```

interface ip pg-communication

Use the **interface ip pg-communication** command to configure the DPE to use the specified interface, identified by its IP address, when communicating with Network Registrar extensions. If you do not specify an interface, the DPE allows the operating system to determine the interface to use while communicating with the Network Registrar extensions.

You can configure either IPv4 address only or both IPv4 and IPv6 addresses by using this command. If IPv4 address is only specified the interface for communication with Network Registrar extensions, the extensions communicate with DPE via the specified IPv4 interface for both IPv4 and IPv6 mode. If both

IPv4 and IPv6 addresses are specified, the extensions communicate with DPE via the specified IPv4 interface in case of IPv4 mode, and the specified IPv6 interface in case of IPv6 mode. IPv6 global address or link local address can be used in the interface ip pg-communication command.

If you do not specify an interface for communication with Network Registrar extensions, the extensions communicate with the DPE via the interface on which provisioning is enabled. If you configure an interface to communicate with the extensions (using the **interface ip pg-communication** command), the extensions communicate with the DPE via the interface you specify. Using this configuration, you can enable the use of split-networking techniques to isolate devices facing communication from management communications.

Note You can configure IPv4/IPv6 interfaces for communication with Network Registrar extensions.

After you use this command, run the **dpe reload** command so that the changes take effect. See dpe reload, page 3-12.

To clear the configured interface, use the **no** form of this command. See no interface ip pg-communication, page 3-18.

Syntax Description

interface ip *ipv4_address* **pg-communication**

ipv4_address—Identifies the IPv4 address of a specific DPE interface.

interface ip *ipv6_address* **pg-communication**

ipv6_address—Identifies the IPv6 address of a specific DPE interface.

Defaults

No default behavior or values.

Examples

This result occurs when you configure an interface by specifying its IPv4 address

```
bac_dpe# interface ip 10.10.1.20 pg-communication
% OK (Requires DPE restart "> dpe reload")
```

This result occurs when you configure an interface by specifying its IPv6 address

```
bac_dpe# interface ip 2001:0DB8:0:0:203:baff:fe12:d5ea pg-communication
% OK (Requires DPE restart "> dpe reload")
```

no interface ip pg-communication

Use the **no interface ip pg-communication** command to disable the interface configured on the DPE when communicating with Network Registrar extensions.

After you use this command, run the **dpe reload** command so that the changes take effect. See dpe reload, page 3-12.

To configure a DPE interface, see interface ip pg-communication, page 3-17.

Syntax Description

no interface ip *ipv4_address* **pg-communication**

ipv4_address—Identifies the IPv4 address of a specific DPE interface.

no interface ip *ipv6_address* **pg-communication**

ipv6_address—Identifies the IPv6 address of a specific DPE interface.

Defaults

No default behavior or values.

Examples

This result occurs when you disable an interface by specifying its IPv4 address

```
bac_dpe# no interface ip 10.10.1.20 pg-communication
% OK (Requires DPE restart "> dpe reload")
```

This result occurs when you disable an interface by specifying its IPv6 address

```
bac_dpe# no interface ip 2001:0DB8:0:0:203:baff:fe12:d5ea pg-communication
% OK (Requires DPE restart "> dpe reload")
```

service tftp

Use the **service tftp** command to configure settings related to TFTP. Table 3-2 lists the keywords that you can use with this command.

The TFTP service on the DPE features one instance of the service, which you can configure to suit your requirements.

Table 3-2 List of service tftp Commands

<table>
<tr><th>Command</th><th>Description</th></tr>
<tr><td>service tftp allow-read-access

no service tftp allow-read-access</td><td>Enables TFTP read requests from the file system. When you enable this command, the DPE looks for the required file in the local directory, and then in the DPE cache.
To disable TFTP read requests from the file system, use the no form of this command.
<table>
<tr><th>Syntax Description</th><th>Defaults</th></tr>
<tr><td>service tftp 1 allow-read-access
1—Identifies the instance of the TFTP service.</td><td>By default, TFTP read requests are disabled.</td></tr>
</table>
Examples
This result occurs when you enable read requests from the file system.
<pre>bac_dpe# service tftp 1 allow-read-access
% OK</pre>This result occurs when you disable read requests from the file system.
<pre>bac_dpe# no service tftp 1 allow-read-access
% OK</pre></td></tr>
</table>

Table 3-2 List of service tftp Commands (continued)

Command	Description
service tftp ipv4 | ipv6 blocksize **no service tftp ipv4 | ipv6 blocksize**	Enables or disables the blocksize option for TFTP transfers using IPv4 or IPv6. The blocksize option specifies the number of data octets and allows the client and server to negotiate a blocksize more applicable to the network medium. When you enable blocksize, the TFTP service uses the requested blocksize for the transfer if it is within the specified lower and upper limits. If you disable blocksize or do not send blocksize option in the TFTP request, the TFTP service uses the 512 blocksize by default. To disable the blocksize option for the TFTP service, use the **no** form of this command. **Note** When the devices, non-compliant with MULPI I09 (or later), request IPv6 blocksize of 1448 instead of 1428, the TFTP request might fail. This failure occurs if the device does not accept the lower negotiated blocksize of 1428; whereas, the upper limit can be configured in the field. There may be an error related to TFTP blocksizes introduced in D3.0 MULPI I09
	Syntax Description **service tftp** *1* **ipv4 | ipv6 blocksize** *lower upper* • *1*—Identifies the instance of the TFTP service. • **ipv4**—Enables blocksize for IPv4. • **ipv6**—Enables blocksize for IPv6. • *lower*—Specifies, in octets, the lower limit of blocksize for the file transfer. If the transfer blocksize is lower than the limit specified, the option is ignored. • *upper*—Specifies, in octets, the upper limit of blocksize for the file transfer. If the transfer blocksize is higher than the limit specified, the option is ignored.
	Defaults By default, the blocksize option is: • Disabled for IPv4. If enabled, the default lower and upper limits are 512 and 1448, respectively. • Enabled for IPv6. The default lower and upper limits are 1428. • If blocksize option is enabled and the requested blocksize is above the maximum, the default upper limit will be used for optimal performance. • If blocksize option is enabled and the requested blocksize is below the minimum, the default lower limit blocksize will be used for optimal performance. • If server is enabled with blocksize option negotiation, the client sends a blocksize option with value within the range of minimum and maximum. The blocksize value can be used for file transfer.

Table 3-2 List of service tftp Commands (continued)

Command	Description

service tftp ipv4 | ipv6 blocksize
 	
no service tftp ipv4 | ipv6 blocksize
	Examples
	This result occurs when you enable blocksize for TFTP transfers.
	• Using IPv4
	`bac_dpe# service tftp 1 ipv4 blocksize 512 1448`
	`% OK`
	• Using IPv6
	`bac_dpe# service tftp 1 ipv6 blocksize 1428 1448`
	`% OK`
	This result occurs when you disable blocksize for TFTP transfers.
	• Using IPv4
	`bac_dpe# no service tftp 1 ipv4 blocksize`
	`% OK`
	• Using IPv6
	`bac_dpe# no service tftp 1 ipv6 blocksize`
	`% OK`

Table 3-2 List of service tftp Commands (continued)

<table>
<tr><th>Command</th><th>Description</th></tr>
<tr><td>service tftp ipv4 | ipv6 enabled</td><td>
Enables or disables the TFTP service for IPv4 or IPv6.

After you run the service tftp command, restart the DPE using the dpe reload command to show the changes. See dpe reload, page 3-12.

Note If the well-known TFTP port (port number 69) is not available, an error message appears.

Syntax Description

service tftp 1 ipv4 | ipv6 enabled true | false

• 1—Identifies the instance of the TFTP service.
• ipv4—Enables the TFTP service for IPv4.
• ipv6—Enables the TFTP service for IPv6.
• true—Enables the TFTP service for IPv4 or IPv6.
• false—Disables the TFTP service for IPv4 or IPv6.

Defaults

The TFTP service is by default disabled.

Examples

This result occurs when you enable the TFTP service.

• For IPv4

<pre>bac_dpe# service tftp 1 ipv4 enabled true
% OK (Requires DPE restart "> dpe reload")</pre>
• For IPv6

<pre>bac_dpe# service tftp 1 ipv6 enabled true
% OK (Requires DPE restart "> dpe reload")</pre>
This result occurs when you disable the TFTP service.

• For IPv4

<pre>bac_dpe# service tftp 1 ipv4 enabled false
% OK (Requires DPE restart "> dpe reload")</pre>
• For IPv6

<pre>bac_dpe# service tftp 1 ipv6 enabled false
% OK (Requires DPE restart "> dpe reload")</pre>
</td></tr>
</table>

Table 3-2 List of service tftp Commands (continued)

Command	Description
service tftp ipv4 | ipv6 verify-ip **no service tftp ipv4 | ipv6 verify-ip**	Enables the verification of requestor IP addresses on dynamic configuration TFTP requests. To disable the verification of requestor IP addresses on dynamic configuration TFTP requests, use the **no** form of this command.
	Syntax Description **service tftp** *1* **ipv4 | ipv6 verify-ip** • *1*—Identifies the instance of the TFTP service. • **ipv4**—Enables verification of requestor IP addresses in IPv4. • **ipv6**—Enables verification of requestor IP addresses in IPv6.
	Defaults The verification of requestor IP addresses on dynamic configuration TFTP requests is by default enabled.

	Examples
	
	This result occurs when you enable verification of requestor IP addresses on TFTP requests.
	
	• For IPv4
	
	`bac_dpe# service tftp 1 ipv4 verify-ip`
	`% OK`
	
	• For IPv6
	
	`bac_dpe# service tftp 1 ipv6 verify-ip`
	`% OK`
	
	This result occurs when you disable verification of requestor IP addresses on TFTP requests.
	
	• For IPv4
	
	`bac_dpe# no service tftp 1 ipv4 verify-ip`
	`% OK`
	
	• For IPv6
	
	`bac_dpe# no service tftp 1 ipv6 verify-ip`
	`% OK`

service tod

Use the **service tod** command to enable or disable the Time of Day (ToD) service running on the DPE for IPv4 or IPv6. The ToD service binds to only those interfaces that are configured for provisioning. For information on how to enable an interface for provisioning, see interface ip provisioning, page 3-14.

After you use this command, run the **dpe reload** command so that the changes take effect. See dpe reload, page 3-12.

Note If the ToD port is not available, an error message appears.

Syntax Description **service tod** *1..1* **ipv4 | ipv6 enabled true | false**

- *1..1*—Identifies the instance of the ToD service.
- **ipv4**—Enables the ToD service for IPv4.
- **ipv6**—Enables the ToD service for IPv6.
- **true**—Enables the ToD service.
- **false**—Disables the ToD service.

Defaults The ToD service is by default disabled on the DPE.

Examples This result occurs when you enable the ToD service on the DPE.

- For IPv4

```
bac_dpe# service tod 1 ipv4 enabled true
% OK (Requires DPE restart "> dpe reload")
```

- For IPv6

```
bac_dpe# service tod 1 ipv6 enabled true
% OK (Requires DPE restart "> dpe reload")
```

This result occurs when you disable the ToD service on the DPE.

- For IPv4

```
bac_dpe# service tod 1 ipv4 enabled false
% OK (Requires DPE restart "> dpe reload")
```

- For IPv6

```
bac_dpe# service tod 1 ipv6 enabled false
% OK (Requires DPE restart "> dpe reload")
```

show device-attribute

Use the **show device-attribute** command to display the last transaction time.

When DPE receives a device configuration request from CNR, it captures the last transaction time as the last seen time of a device.

Note This feature might utilize about 1 to 1.5 GB disk space on BPR_DATA directory of DPE.

Syntax Description **show device-attribute** last-seen-time *mac | duid*

- *mac*—Specifies the MAC address of a device. The accepted formats for *mac,* assuming that the MAC address header is 1,6, are:
  - "Type,len,addr"; for example, 1,6,00:01:02:03:04:05 or 9,10,43:43:31:32:33:34:35:36:2d:41.
  - Exact-size octets; for example, 000102030405 or 00:01:02:03:04:05.

- *duid*—Specifies the DHCP Unique Identifier (DUID) of a device in an IPv6 environment; for example, 00:03:00:01:00:18:68:52:75:c0. A DUID cannot be more than 128 octets long.

Defaults No default behavior or values.

Examples

- For IPV4 device using MAC address

```
bac_dpe# show device-attributes mac 1,6,00:00:00:00:08:09
Fetching attributes for device [1,6,00:00:00:00:08:09]
last-seen-time : 1478077900666
```

- For IPV6 device using duid

```
bac_dpe# show device-attribute last-seen-time duid 00:03:00:01:00:00:00:00:05:07
Fetching attribute [last-seen-time] for device [00:03:00:01:00:00:00:00:05:07]
Attribute(s) does not available for device [00:03:00:01:00:00:00:00:05:07]
```

dump device-attributes

Use this command to dump all the device attributes from a DPE. This information is exported as a *.csv* file and is stored as device_attributes.csv file in the following path:

BPR_DATA/dpe/cache/device_attributes.csv

Syntax Description **dump device-attributes**

Defaults No default behavior or values.

Examples This result occurs when you dump all the device attributes from a DPE:

```
bac_dpe# dump device-attributes
Sending request to dump device attributes...
Initiated the request for dumping device attributes. Device attributes will be exported to
a CSV file [/var/CSCObac/dpe/cache/device_attributes.csv]
```

show dump-device-attributes-status

Use the **show dump-device-attributes** command to know the status of the dumping process.

Syntax Description **show dump-device-attributes-status**

Defaults No default behavior or values.

Examples This result occurs when you want to see the status of the device attributes dumping:

```
bac_dpe# show dump-device-attributes-status
There is no dumping process currently running.
```

show device-config

Use the **show device-config** command to display a device configuration that is cached at the DPE.

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 **show device-config** *mac* | *duid*

- *mac*—Specifies the MAC address of a device. The accepted formats for *mac,* assuming that the MAC address header is 1,6, are:
  - "Type,len,addr"; for example, 1,6,00:01:02:03:04:05 or 9,10,43:43:31:32:33:34:35:36:2d:41.
  - Exact-size octets; for example, 000102030405 or 00:01:02:03:04:05.
- *duid*—Specifies the DHCP Unique Identifier (DUID) of a device in an IPv6 environment; for example, 00:03:00:01:00:18:68:52:75:c0. A DUID cannot be more than 128 octets long.

Defaults No default behavior or values.

Examples This result occurs when you look up a configuration based on the MAC address of the device. This example assumes that the MAC address is 1,6,aa:bb:cc:dd:ee:ff.

```
bac_dpe# show device-config mac 1,6,aa:bb:cc:dd:ee:ff
DHCP configuration for device 1,6,aa:bb:cc:dd:ee:ff in default provisioning-group:
  Extension PRE_CLIENT_LOOKUP
    Dictionary REQUEST
      VALIDATE relay-agent-remote-id = 00:00:00:00:aa:bb:cc:dd
      VALIDATE_CONTINUE dhcp-parameter-request-list-blob =
42:43:01:03:02:04:07:06:0c:0f:7a:b1
      VALIDATE_CONTINUE dhcp-class-identifier =
"docsis1.1:052401010102010103010104010105010106010107010f0801100901000a01010b01080c0101"
    Dictionary ENVIRONMENT
      PUT_REPLACE client-class-name = "unprovisioned-docsis"
  Extension PRE_PACKET_ENCODE
    Dictionary RESPONSE
      PUT_REPLACE ccc-primary-dhcp-server = BYTES_BPR_PROPERTY_OPTIONAL_IP_ADDRESS_BIN
"/ccc/dhcp/primary"
      PUT_REPLACE ccc-secondary-dhcp-server = BYTES_BPR_PROPERTY_OPTIONAL_IP_ADDRESS_BIN
"/ccc/dhcp/secondary"
      PUT_REPLACE boot-file = "unprov.cm"
      PUT_REPLACE file = "unprov.cm"
      PUT_REPLACE siaddr = BYTES_DPE_IP_ADDRESS_BIN
      PUT_REPLACE tftp-server = BYTES_DPE_IP_ADDRESS_DOTTED_DECIMAL
      PUT_REPLACE time-servers = BYTES_DPE_IP_ADDRESS_BIN
```

This result occurs when you look up a configuration based on the DUID of the device. This example assumes that the DUID is 00:00:00:00:00:00:00:52:75:c0.

```
bac_dpe# show device-config duid 00:00:00:00:00:00:00:52:75:c0
DHCP configuration for device 00:00:00:00:00:00:00:52:75:c0 in default provisioning-group:
DHCP Configuration for device 00:00:00:00:00:00:00:52:75:c0
    Commands:
        PRE_CLIENT_LOOKUP: ENVIRONMENT, PUT_REPLACE, client-class-name,
unprovisioned-docsis
        PRE_CLIENT_LOOKUP: RELAY_REQUEST, VALIDATE_CONTINUE, link-address,
20:01:04:20:38:00:05:00:00:00:00:00:00:00:00:01
        PRE_CLIENT_LOOKUP: REQUEST, VALIDATE_OPTION_CONTINUE, {OPTION_NUMBER=16,
ENTERPRISE_ID=4491, INDEX=0, END}, 64:6f:63:73:69:73:33:2e:30
        PRE_PACKET_ENCODE: RESPONSE, PUT_OPTION, {OPTION_NUMBER=17, ENTERPRISE_ID=4491,
SUBOPTION_NUMBER=33, END}, unprov.cm
        PRE_PACKET_ENCODE: RESPONSE, PUT_OPTION, {OPTION_NUMBER=17, ENTERPRISE_ID=4491,
SUBOPTION_NUMBER=37, END}, BYTES_DPE_IPV6_ADDRESS_BIN
        PRE_PACKET_ENCODE: RESPONSE, PUT_OPTION, {OPTION_NUMBER=17, ENTERPRISE_ID=4491,
SUBOPTION_NUMBER=32, END}, BYTES_DPE_IPV6_ADDRESS_BIN
```

This result occurs when the configuration for the specified device is not available in the DPE cache.

```
bac_dpe# show device-config mac 1,6,aa:bb:cc:dd:ee:aa
No configuration found on DPE.
```

show dpe

Use the **show dpe** command to check to see if the DPE is running and to display the state of the process and, if running, its operational statistics. This command does not indicate if the DPE is running successfully, only that the process itself is currently executing. However, when the DPE is running, you can use statistics that this command displays to determine if the DPE is successfully servicing requests.

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

No keywords or arguments.

Defaults

No default behavior or values.

Examples

This result occurs when the DPE is running.

```
bac_dpe# show dpe
Process [dpe] is running

Version BAC 4.0 (SOL_BAC5_0_0_20000000_0000).
Caching 0 device configs and 6 external files.
Received 0 cache hits and 3 misses.
Received 0 lease updates.
Connection status is Ready.
Sent 0 SNMP informs and 0 SNMP sets.
Received 0 MTA provisioning successful SNMP informs.
Received 0 MTA provisioning failed SNMP informs.
Running for 10 hours 51 mins 23 secs.
```

This result occurs when the DPE is not running.

```
bac_dpe# show dpe
BAC Process Watchdog is running
Process [dpe] is not running
```

When this error occurs, start the DPE process. See dpe start | stop, page 3-13.

This result occurs when the DPE is unable to service requests.

```
bac_dpe# show dpe
BAC Process Watchdog is running
Process [dpe] is not running; it is in back off mode
```

This error occurs when there is an issue with the DPE. Look at the DPE log (*dpe.log*) to troubleshoot the issue.

show dpe config

Use the **show dpe config** command to display the current settings on the DPE.

Syntax Description

No keywords or arguments.

Defaults

No default behavior or values.

Examples

```
bac_dpe# show dpe config
dpe port         = 49186
rdu host         = source
rdu port         = ip
primary groups   = provisioning-second-1,provisioning-second-2
secondary groups = [no value]
```

PacketCable Voice Technology Commands

This chapter describes the command-line interface (CLI) commands that you can use to manage and monitor the PacketCable voice technology on the Prime Cable Provisioning Device Provisioning Engine (DPE).

The commands described in this chapter are:

Command	Description	CLI Mode		Required Privileges			
		Login	Enable	PRIV_DPE_READ	PRIV_DPE_UPDATE	PRIV_DPE_SECURITY	PRIV_DEVICE_READ
debug service packetcable netsnmp	Enables the PacketCable NetSNMP category for debug messages.		✓	✓	✓		
debug service packetcable registration	Enables the PacketCable registration category for debug messages.		✓	✓	✓		
debug service packetcable registration-detail	Enables the PacketCable registration detail category for debug messages.		✓	✓	✓		
debug service packetcable snmp	Enables the PacketCable SNMP service category for debug messages.		✓	✓	✓		
service packetcable enable	Enables or disables the PacketCable services.		✓	✓	✓		
service packetcable registration encryption enable	Enables encryption on MTA configuration files.		✓	✓	✓		

Command	Description	CLI Mode		Required Privileges			
		Login	Enable	PRIV_DPE_READ	PRIV_DPE_UPDATE	PRIV_DPE_SECURITY	PRIV_DEVICE_READ
service packetcable registration kdc-service-key	Sets the service key for KDC communications.		✓	✓	✓	✓	
service packetcable registration policy-privacy	Sets the customer policy regarding enforcement of SNMP privacy in MTA communications.		✓	✓	✓		
service packetcable snmp key-material	Sets the key material for MTA SNMP communications.		✓	✓	✓	✓	
service packetcable snmp timeout	Sets the timeout value for SNMP SET operations.		✓	✓	✓		
service packetcable show snmp log	Displays PacketCable SNMP log entries.		✓	✓	✓		

debug service packetcable

Use the **debug service packetcable** command to debug the PacketCable technology service on the DPE. Table 4-1 lists the keywords that you can use with this command. The PacketCable service on the DPE features one instance of the service, which you can configure to suit your requirements.

Before using any debug command, you must enable debugging by running the **debug on** command. 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.
```

Caution Enabling debug logging may have a severe impact on DPE performance. Do not leave the DPE running with debug turned on for an extended period of time.

Table 4-1 List of debug service packetcable Commands for PacketCable Technology

Command	Description	
debug service packetcable netsnmp **no debug service packetcable netsnmp**	Enables detailed debugging of the PacketCable NetSNMP service on the DPE. To disable detailed debugging of the PacketCable NetSNMP service, use the **no** form of this command.	
	Syntax Description **debug service packetcable** *1..1* **netsnmp** *1..1*—Identifies the instance of the PacketCable service.	**Defaults** Debugging of the PacketCable NetSNMP service is by default disabled.

	Examples
	
	This result occurs when you enable debugging of the PacketCable NetSNMP service.
	
	`bac_dpe# debug service packetcable 1 netsnmp`
	`% OK`
	
	This result occurs when you disable debugging of the PacketCable NetSNMP service.
	
	`bac_dpe# no debug service packetcable 1 netsnmp`
	`% OK`	

debug service packetcable registration **no debug service packetcable registration**	Enables debugging of the PacketCable secure registration service on the DPE. To disable debugging of the PacketCable secure registration service, use the **no** form of this command.	
	Syntax Description **debug service packetcable** *1..1* **registration** *1..1*—Identifies the instance of the PacketCable service.	**Defaults** Debugging of the PacketCable registration service is by default disabled.

	Examples
	
	This result occurs when you enable debugging of the PacketCable registration service.
	
	`bac_dpe# debug service packetcable 1 registration`
	`% OK`
	
	This result occurs when you disable debugging of the PacketCable registration service.
	
	`bac_dpe# no debug service packetcable 1 registration`
	`% OK`	

Table 4-1 List of debug service packetcable Commands for PacketCable Technology (continued)

Command	Description	
debug service packetcable registration-detail **no debug service packetcable registration-detail**	Enables the PacketCable registration detail category for debug messages. To disable debugging of the PacketCable secure registration service, use the **no** form of this command.	
	Syntax Description **debug service packetcable** *1..1* **registration-detail** *1..1*—Identifies the instance of the PacketCable service.	**Defaults** Debugging of the PacketCable registration detail category is by default disabled.

	Examples
	This result occurs when you enable debugging of the PacketCable registration detail category.
	`bac_dpe# debug service packetcable 1 registration-detail`
	`% OK`
	
	This result occurs when you disable debugging of the PacketCable registration detail category.
	`bac_dpe# no debug service packetcable 1 registration-detail`
	`% OK`	

debug service packetcable snmp **no debug service packetcable snmp**	Enables detailed debugging of the PacketCable SNMP service on the DPE. To disable detailed debugging of the PacketCable SNMP service, use the **no** form of this command.	
	Syntax Description **debug service packetcable** *1..1* **snmp** *1..1*—Identifies the instance of the PacketCable service.	**Defaults** Debugging of the PacketCable SNMP service is by default disabled.

	Examples
	This result occurs when you enable debugging of the PacketCable SNMP service.
	`bac_dpe# debug service packetcable 1 snmp`
	`% OK`
	
	This result occurs when you disable debugging of the PacketCable SNMP service.
	`bac_dpe# no debug service packetcable 1 snmp`
	`% OK`	

service packetcable enable

Use the **service packetcable enable** command to enable the PacketCable service on the DPE.

To enable PacketCable, you must:

- Configure at least one interface with a fully qualified domain name (FQDN) and enable provisioning. See interface ip provisioning fqdn, page 3-16, and interface ip provisioning, page 3-14.

  If you do not configure an interface with an FQDN and enable provisioning on that interface, the following error appears:

  ```
  Enabling packetcable requires at least one interface must have an FQDN configured and provisioning enabled
  Error processing command
  ```

- Set the service key for the Key Distribution Center (KDC). See service packetcable registration kdc-service-key, page 4-7.

Note To enable PacketCable the kdc-service key must match the dpe-service key.

If you do not set a service key for the KDC, the following error appears:

```
A KDC service key must be present in order to enable PacketCable

Error processing command
```

After you use this command, run the **dpe reload** command so that the changes take effect. See dpe reload, page 3-12.

Syntax Description

service packetcable *1..1* **enable**

1..1—Identifies the instance of the PacketCable service.

Defaults

The PacketCable service on the DPE is by default enabled.

Examples

```
bac_dpe# service packetcable 1 enabled true
% OK (Requires DPE restart "> dpe reload")
```

no service packetcable enable

Use the **no service packetcable enable** command to disable the PacketCable service on the DPE.

Syntax Description

no service packetcable *1..1* **enable**

1..1—Identifies the instance of the PacketCable service.

Defaults The PacketCable service on the DPE is by default enabled.

Examples

```
bac_dpe# no service packetcable 1
% OK (Requires DPE restart "> dpe reload")
```

service packetcable registration encryption enable

Use the **service packetcable registration encryption enable** command to enable encryption of MTA configuration files.

To disable encryption of MTA configuration files, use the **no** form of this command. See no service packetcable registration encryption, page 4-6.

Syntax Description **service packetcable** *1..1* **registration encryption enable**

1..1—Identifies the instance of the PacketCable service.

Defaults Encryption of MTA configuration files is by default disabled.

Examples

```
bac_dpe# service packetcable 1 registration encryption enable
% OK
```

no service packetcable registration encryption

Use the **no service packetcable registration encryption** command to disable encryption of MTA configuration files.

To enable encryption of MTA configuration files, see service packetcable registration encryption enable, page 4-6.

Syntax Description **no service packetcable** *1..1* **registration encryption**

1..1—Identifies the instance of the PacketCable service.

Defaults Encryption of MTA configuration files is by default disabled.

Examples

```
bac_dpe# no service packetcable 1 registration encryption
% OK
```

service packetcable registration kdc-service-key

Use the **service packetcable registration kdc-service-key** command to generate and set a security key for communication between the KDC and a DPE.

After you use this command, run the **dpe reload** command so that the changes take effect. See dpe reload, page 3-12.

Syntax Description

service packetcable *1..1* **registration kdc-service-key** *password*

- *1..1*—Identifies the instance of the PacketCable service.
- *password*—Identifies the password, which must be from 6 to 20 characters.

Note The password that you enter must match the password that you enter while configuring the KDC using the KeyGen tool. See the *Cisco Prime Cable Provisioning 6.1.3 User Guide* for information on how to use the KeyGen tool.

You can verify the service key that this command creates by viewing the *dpe.properties* file, which resides in the *BPR_HOME/dpe/conf* directory. Look for the value of the following parameter: `/pktcbl/regsvr/KDCServiceKey`.

For example:

```
# more dpe.properties
...
/pktcbl/regsvr/KDCServiceKey=2e:d5:ef:e9:5a:4e:d7:06:67:dc:65:ac:bb:89:e3:2c:bb:
71:5f:22:bf:94:cf:2c
...
```

The output of this example is trimmed.

Defaults

No default behavior or values.

Examples

```
bac_dpe# service packetcable 1 registration kdc-service-key password3
% OK (Requires DPE restart "> dpe reload")
```

service packetcable registration policy-privacy

Use the **service packetcable registration policy-privacy** command to set the customer policy on enforcing SNMP privacy in MTA communications.

Entering a value of zero lets the MTA choose the SNMPv3 privacy option. Entering a nonzero value means that the provisioning server sets the privacy option in SNMPv3 to a specific protocol, which is currently limited to DES.

After you use this command, run the **dpe reload** command so that the changes take effect. See dpe reload, page 3-12.

Syntax Description

service packetcable *1..1* **registration policy-privacy** *value*

- *1..1*—Identifies the instance of the PacketCable service.
- *value*—Enter any zero or nonzero value to identify the customer policy. Values include:
  - 0—Indicates that the MTA selects the privacy option with Privacy being optional.
  - 1—Indicates that the policy is enforced, causing all MTAs to use Privacy. If Privacy is not used, the MTA does not start.
  - 32—Indicates that there is no Privacy.
  - 33—Indicates that Privacy is enabled for all devices.

Defaults

The default value for enforcing SNMP privacy is 1.

Examples

This result occurs when you enforce SNMP privacy, using the default value of 1, causing all MTAs to use Privacy.

```
bac_dpe# service packetcable 1 registration policy-privacy 1
% OK (Requires DPE restart "> dpe reload">
```

service packetcable snmp key-material

Use the **service packetcable snmp key-material** command to generate and set a security key on the DPE to permit secure communication with the RDU. The secure communication channel with the RDU is used for PacketCable SNMPv3 cloning support only.

Note You must set the same security key on both the DPE and the RDU. Use the **generateSharedSecret.sh** command-line tool, located in the *BPR_HOME/rdu/bin* directory.

After you use this command, run the **dpe reload** command so that the changes take effect. See dpe reload, page 3-12.

To clear the SNMPv3 service key and turn off the SNMPv3 cloning support, use the **no** form of this command. See no service packetcable snmp key-material, page 4-9.

Syntax Description

service packetcable *1..1* **snmp key-material** *password*

- *1..1*—Identifies the instance of the PacketCable service.
- *password*—Identifies the password that you create, which must be from 6 to 20 characters.

Defaults

Generating a security key for secure communication with the RDU is by default disabled.

Examples

```
bac_dpe# service packetcable 1 snmp key-material password4
% OK (Requires DPE restart "> dpe reload")
```

no service packetcable snmp key-material

Use the **no service packetcable snmp key-material** command to clear the SNMPv3 service key and turn off SNMPv3 cloning support.

After you use this command, run the **dpe reload** command so that the changes take effect. See dpe reload, page 3-12.

To generate and set a security key on the DPE for secure communication with the RDU, see service packetcable snmp key-material, page 4-8.

Syntax Description

no service packetcable *1..1* **snmp key-material**

1..1—Identifies the instance of the PacketCable service.

Defaults

Generating a security key for secure communication with the RDU is by default disabled.

Examples

```
bac_dpe# no service packetcable 1 snmp key-material
% OK (Requires DPE restart "> dpe reload")
```

service packetcable snmp timeout

Use the **service packetcable snmp timeout** command to dynamically set the length of time that the PacketCable SNMP service waits for a response to any SNMP ‘Set’ operation.

Syntax Description

service packetcable *1..1* **snmp timeout** *time*

- *1..1*—Identifies the instance of the PacketCable service.
- *time*—Indicates the length of time that the PacketCable SNMP service waits, in seconds.

Defaults

The default maximum length of time that the PacketCable SNMP service waits for a response to an SNMP ‘Set’ operation is 10 seconds.

Examples

```
bac_dpe# service packetcable 1 snmp timeout 15
% OK
```

service packetcable show snmp log

Use the **service packetcable show snmp log** command to show recent log entries for the PacketCable SNMP provisioning service, which includes information about the general PacketCable SNMP provisioning service and the logging of any MTA provisioning errors or severe problems.

Syntax Description

service packetcable *1..1* **show snmp log** [**last** *1..9999* | **run**]

- *1..1*—Identifies the instance of the PacketCable service.
- **last** *1..9999*—Identifies the specified number of recent log entries from the PacketCable SNMP log file that you want to display. This keyword is optional.
- **run**—Displays all log messages from the PacketCable SNMP log file. This keyword is optional.

Defaults

No default behavior or values.

Examples

This result occurs when you use the **service packetcable show snmp log** command to display all log entries for the PacketCable SNMP service.

```
bac_dpe# service packetcable 1 show snmp log
Error [SS_MSG] 2007-12-18 14:30:44,000 - SNMP Service Tracing Set To 400
...
```

Note The output presented in this example is trimmed.

This result occurs when you use the **service packetcable show snmp log last** command to display a specific number of recent log entries; in this example, the last 5 entries.

```
bac_dpe# service packetcable 1 show snmp log last 5
Error [SS_MSG] 2007-12-18 14:35:44,000 - SNMP Service Tracing Set To 800
```

This result occurs when you use the **service packetcable show snmp log run** command to display a running PacketCable SNMP log. The command continues to run until you press **Enter**.

```
bac_dpe # service packetcable 1 show snmp log run
Press <enter> to stop.

2007 12 17 11:43:43 CDT: %CSRC-5: Notification DPE: Device Provisioning Engine starting up
2007 12 17 11:43:44 CDT: %CSRC-6: Info DPE: Attempt to connect to RDU dpe failed;
2007 12 17 11:43:44 CDT: %CSRC-6: Info TFTP: Ready to service requests

Stopped.
```

SNMP Agent Commands

This chapter describes the command-line interface (CLI) commands that you can use to manage and monitor the SNMP agent on the Prime Cable Provisioning Device Provisioning Engine (DPE).

The commands described in this chapter are:

Command	Description	CLI Mode		Required Privileges			
		Login	Enable	PRIV_DPE_READ	PRIV_DPE_UPDATE	PRIV_DPE_SECURITY	PRIV_DEVICE_READ
snmp-server community	Defines the community string.		✓	✓	✓		
no snmp-server community	Clears the specified community string.		✓	✓	✓		
snmp-server contact	Sets the system contact.		✓	✓	✓		
no snmp-server contact	Clears the specified system contact.		✓	✓	✓		
snmp-server host	Sets the SNMP notification recipient host.		✓	✓	✓		
no snmp-server host	Clears the SNMP notification recipient host.		✓	✓	✓		
snmp-server inform	Sets the notification type to inform.		✓	✓	✓		
no snmp-server inform	Sets the notification type to trap.		✓	✓	✓		
snmp-server location	Sets system location.		✓	✓	✓		

Command	Description	CLI Mode		Required Privileges			
		Login	Enable	PRIV_DPE_READ	PRIV_DPE_UPDATE	PRIV_DPE_SECURITY	PRIV_DEVICE_READ
no snmp-server location	Clears system location.		✓	✓	✓		
snmp-server reload	Restarts the SNMP processes.		✓	✓	✓		
snmp-server start | stop	Starts or stops the SNMP processes.		✓	✓	✓		
snmp-server udp-port	Sets the UDP port to which the SNMP agent listens.		✓	✓	✓		
no snmp-server udp-port	Sets the configured UDP port to which the SNMP agent listens back to the default port.		✓	✓	✓		

snmp-server community

Use the **snmp-server community** command to define the community string that allows external SNMP managers access to the SNMP agent on the DPE.

After you use this command, run the **snmp-server reload** command so that the changes take effect. See snmp-server reload, page 5-7.

To delete the specified community string, use the **no** form of this command. See no snmp-server community, page 5-3.

Syntax Description

snmp-server community *string* [**ro** | **rw**]

- *string*—Identifies the SNMP community.
- **ro**—Assigns a read-only community string. Only Get requests (queries) can be performed. The network management system and the managed device must reference the same community string.
- **rw**—Assigns a read-write community string. SNMP applications require **rw** access for Set operations. The **rw** community string enables write access to vendor ID values.

Defaults

The default **ro** and **rw** community strings are **baccread** and **baccwrite**, respectively. We recommend that you change these values before deploying Prime Cable Provisioning.

Examples

This result occurs when you use the default **baccread** option for the read-only community string.

```
bac_dpe# snmp-server community baccread ro
```

```
% OK ()
Requires SNMP agent restart "> snmp-server reload"
```

This result occurs when you use the default **baccwrite** option for the read-write community string.

```
bac_dpe# snmp-server community baccwrite rw
% OK ()
Requires SNMP agent restart "> snmp-server reload"
```

no snmp-server community

Use the **no snmp-server community** command to delete the specified community string that allows access for external SNMP managers to the SNMP agent on the DPE.

After you use this command, run the **snmp-server reload** command to restart the SNMP agent. See snmp-server reload, page 5-7.

To set up the community access string, see snmp-server community, page 5-2.

Syntax Description

no snmp-server community *string*

string—Identifies the SNMP community.

Defaults

No default behavior or values.

Examples

```
bac_dpe# no snmp-server community test_community
% OK ()
Requires SNMP agent restart "> snmp-server reload"
```

snmp-server contact

Use the **snmp-server contact** command to enter a string of characters that identify the system contact (sysContact) as defined in the MIB II.

After you use this command, run the **snmp-server reload** command to restart the SNMP agent. See snmp-server reload, page 5-7.

To remove the system contact, use the **no** form of this command. See no snmp-server contact, page 5-4.

Syntax Description

snmp-server contact *text*

text—Identifies the name of the contact responsible for the DPE.

Defaults

No default behavior or values.

Examples

```
bac_dpe# snmp-server contact joe
```

```
% OK (Requires SNMP server restart "> snmp-server reload")
```

no snmp-server contact

Use the **no snmp-server contact** command to remove the system contact that is responsible for the DPE.

After you use this command, run the **snmp-server reload** command to restart the SNMP agent. See snmp-server reload, page 5-7.

To enter a string of characters that identify the system contact, use the **snmp-server contact** command. See snmp-server contact, page 5-3.

Syntax Description No keywords or arguments.

Defaults No default behavior or values.

Examples

```
bac_dpe# no snmp-server contact
% OK (Requires SNMP server restart "> snmp-server reload")
```

snmp-server host

Use the **snmp-server host** command to specify the recipient of all SNMP notifications and to configure the SNMP agent to send traps or informs to multiple hosts.

Note You can use multiple instances of this command to specify more than one notification recipient.

After you use this command, run the **snmp-server reload** command so that the changes take effect. See snmp-server reload, page 5-7.

To remove the specified notification recipient, use the **no** form of this command. See no snmp-server host, page 5-5.

Syntax Description **snmp-server host** *host-addr* **notification community** *community* [**udp-port** *port*]

- *host-addr*—Specifies the IP address of the host to which notifications are sent.
- *community*—Specifies the community string to use while sending SNMP notifications.
- *port*—Identifies the UDP port used to send SNMP notifications. The default port number is 162.

Defaults No default behavior or values.

Examples

```
bac_dpe# snmp-server host 10.10.10.5 notification community public udp-port 162
% OK ()
```

```
Requires SNMP agent restart "> snmp-server reload"
```

no snmp-server host

Use the **no snmp-server host** command to remove the specified notification recipient.

After you use this command, run the **snmp-server reload** command so that the changes take effect. See snmp-server reload, page 5-7.

To specify the recipient of all SNMP notifications, see snmp-server host, page 5-4.

Syntax Description

no snmp-server host *host-add* **notification**

host-add—Identifies the IP address of the host.

Defaults

No default behavior or values.

Examples

```
bac_dpe# no snmp-server host 10.10.10.5 notification
% OK ()
Requires SNMP agent restart "> snmp-server reload"
```

snmp-server inform

Use the **snmp-server inform** command to specify the type of SNMP notification sent from the SNMP agent to the SNMP manager. Use it to send SNMP informs rather than traps, although traps are sent by default.

After you use this command, run the **snmp-server reload** command to restart the SNMP agent. See snmp-server reload, page 5-7.

To switch the SNMP notifications back to the default setting of traps, use the **no** form of this command. See no snmp-server inform, page 5-6.

Syntax Description

snmp-server inform [**retries** *count* **timeout** *time*]

- *count*—Identifies the number of times an inform can be sent from the SNMP agent to the manager. If the timeout period expires before the configured number of retries is reached, the SNMP server stops sending informs.
- *time*—Identifies the length of time (in milliseconds) that the SNMP server continues to send informs. If the maximum number of retries is reached before the timeout expires, the SNMP server stops sending informs.

Note Specifying the retry count and the timeout while configuring SNMP informs is optional. If you do not specify any values, the default values are used.

Defaults SNMP notification via informs is by default disabled. If you configure SNMP notification as informs, the default number of retries is 1 and the default timeout is 5000 milliseconds.

Examples In this example, an SNMP inform will be sent up to a maximum of five times before the retries stop. If the timeout of 500 milliseconds expires before the five retries take place, the inform is not sent again.

```
bac_dpe# snmp-server inform retries 5 timeout 500
% OK ()
Requires SNMP agent restart "> snmp-server reload"
```

no snmp-server inform

Use the **no snmp-server inform** command to switch the SNMP notifications that are sent to the SNMP manager back to the default setting of traps.

After you use this command, run the **snmp-server reload** command to restart the SNMP agent. See snmp-server reload, page 5-7.

To specify the type of SNMP notification sent, see snmp-server inform, page 5-5.

Syntax Description No keywords or arguments.

Defaults SNMP notification is by default set to traps (not informs).

Examples

```
bac_dpe# no snmp-server inform
% OK ()
Requires SNMP agent restart "> snmp-server reload"
```

snmp-server location

Use the **snmp-server location** command to enter a string of characters that identify the system location (sysLocation) as defined in the MIB II.

After you use this command, run the **snmp-server reload** command to restart the SNMP agent. See snmp-server reload, page 5-7.

To remove a system location, use the **no** form of this command. See no snmp-server location, page 5-7.

Syntax Description **snmp-server location** *text*

text—Identifies the physical location of the DPE.

Defaults No default behavior or values.

Examples

```
bac_dpe# snmp-server location st_louis
% OK (Requires SNMP agent restart "> snmp-server reload")
```

no snmp-server location

Use the **no snmp-server location** command to remove a system location.

After you use this command, run the **snmp-server reload** command to restart the SNMP agent. See snmp-server reload, page 5-7.

To enter a string of characters that identify the system location, see snmp-server location, page 5-6.

Syntax Description

No keywords or arguments.

Defaults

No default behavior or values.

Examples

```
bac_dpe# no snmp-server location
% OK (Requires SNMP server restart "> snmp-server reload")
```

snmp-server reload

Use the **snmp-server reload** command to reload the SNMP agent process on the DPE.

Note When the SNMP process is started on the RDU and DPE, a trap containing the system uptime is sent. Prime Cable Provisioning trap notifications, however, are disabled by default. You can enable trap notifications only by setting the corresponding MIB object via SNMP. You cannot enable trap notifications via the CLI or the Admin UI.

This Prime Cable Provisioning release supports only the trap notifications defined in the CISCO-BACC-SERVER-MIB and CISCO-BACC-RDU-MIB files. For more information, see the MIB files in the *BPR_HOME/rdu/mibs* directory.

Syntax Description

No keywords or arguments.

Defaults

No default behavior or values.

Examples

```
bac_dpe# snmp-server reload
Process [snmpAgent] has been restarted.

bac_dpe#
```

snmp-server start | stop

Use the **snmp start | stop** command to start or stop the SNMP agent process on the DPE.

Syntax Description

snmp-server start | stop

- **start**—Starts the SNMP agent process on the DPE.
- **stop**—Stops the SNMP agent process on the DPE.

Defaults

No default behavior or values.

Examples

This result occurs when the SNMP agent process is started.

```
bac_dpe# snmp-server start
Process [snmpAgent] has been started.

bac_dpe#
```

This result occurs when the SNMP agent process is already running.

```
bac_dpe# snmp-server start
Process [snmpAgent] is already running
```

This result occurs when the SNMP agent process is stopped.

```
bac_dpe# snmp-server stop
Process [snmpAgent] has been stopped.

bac_dpe#
```

snmp-server udp-port

Use the **snmp-server udp-port** command to identify the UDP port number on which the SNMP agent listens.

The DPE requires this command to prevent potential sharing violations between ports that other applications use. The changing of port numbers is used to resolve potential port conflict.

To change the port to which the SNMP agent listens back to the default UDP port number, use the **no** form of this command. See no snmp-server udp-port, page 5-9.

Syntax Description

snmp-server udp-port *port*

port—Identifies the UDP port to which the SNMP agent listens.

Defaults

The default port number of the SNMP agent is 8001.

Note To eliminate potential port conflicts with other SNMP agents on the computer, the default port number is different from the standard well-known SMNP agent port.
We recommend that you change the SNMP agent port to the well-known port number 161.

Examples

```
bac_dpe# snmp-server udp-port 161
% OK ()
Requires SNMP agent restart "> snmp-server reload"
```

no snmp-server udp-port

Use the **no snmp-server udp-port** command to change the UDP port to which the SNMP agent listens to the default port (8001).

Note Using a port number other than the standard well-known SNMP agent port number of 161 increases the likelihood of potential port conflicts with other SNMP agents running on the same computer.

To specify the UDP port number to which the SNMP agent listens, see snmp-server udp-port, page 5-8.

Syntax Description No keywords or arguments.

Defaults The default port number of the SNMP agent is 8001.

Examples

```
bac_dpe# no snmp-server udp-port
% OK ()
Requires SNMP agent restart "> snmp-server reload"
```

Log System Management Commands

This chapter describes the command-line interface (CLI) commands that you can use to debug the Prime Cable Provisioning Device Provisioning Engine (DPE), and monitor and manage the Prime Cable Provisioning log system.

Before using a debug command, you must enable DPE debugging by running the **debug on** command. 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 a DPE license.
```

Caution Enabling debug logging may have a severe impact on DPE performance. Do not leave the DPE running with debug turned on for an extended period of time.

The commands described in this chapter are:

Command	Description	CLI Mode		Required Privileges			
		Login	Enable	PRIV_DPE_READ	PRIV_DPE_UPDATE	PRIV_DPE_SECURITY	PRIV_DEVICE_READ
clear logs	Removes out-of-date log files from the system.		✓	✓	✓		
debug dpe cache	Debugs the DPE cache.		✓	✓	✓		
debug dpe connection	Debugs the DPE connection.		✓	✓	✓		
debug dpe dpe-server	Debugs the DPE server.		✓	✓	✓		
debug dpe event-manager	Debugs the DPE event manager.		✓	✓	✓		
debug dpe exceptions	Debugs DPE exceptions.		✓	✓	✓		
debug dpe framework	Debugs the DPE framework.		✓	✓	✓		

Command	Description	CLI Mode		Required Privileges			
		Login	Enable	PRIV_DPE_READ	PRIV_DPE_UPDATE	PRIV_DPE_SECURITY	PRIV_DEVICE_READ
debug dpe messaging	Debugs DPE messaging.		✓	✓	✓		
debug dpe ssl_all	Enables the JSSE internal messaging category for debugging ssl messages		✓	✓	✓		
debug dpe secure_messaging	Enables the ssl messaging category for debug messages		✓	✓	✓		
debug on	Enables debug logging.		✓	✓	✓		
debug service tftp ipv4 | ipv6	Debugs TFTP transfers.		✓	✓	✓		
no debug all	Disables debug logging.		✓	✓	✓		
log level	Sets the level of minimum DPE log messages.		✓	✓	✓		
show log	Displays recent log entries for the DPE.	✓	✓	✓			

clear logs

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

- DPE logs
- Hardware
- Syslog

Over time, historic log files accumulate within the DPE. You can use the **support bundle state** command to bundle these logs. We recommend that you create a bundle before clearing logs, so that no necessary files are lost accidently.

Syntax Description

No keywords or arguments.

Defaults No default behavior or values.

Examples

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

debug dpe

Use the **debug dpe** command to configure debug settings on the DPE. Table 6-1 describes the keywords that you can use with this command.

Note Enter the commands described in Table 6-1 as indicated.

Table 6-1 List of debug dpe Commands

Command	Description	
debug dpe cache **no debug dpe cache**	Enables debugging of the DPE cache, which involves messages pertaining to the DPE cache including: • Logging requests for cache entries • Updates to the cache • Other interactions by DPE subsystems To disable DPE cache debugging, use the **no** form of this command.	

	Examples
	This result occurs when you enable debugging of the DPE cache.
	`bac_dpe# debug dpe cache`
	`% OK`
	
	This result occurs when you disable debugging of the DPE cache.
	`bac_dpe# no debug dpe cache`
	`% OK`
		Defaults
		Debugging of the DPE cache is by default disabled.

Table 6-1 List of debug dpe Commands (continued)

Command	Description	
debug dpe connection **no debug dpe connection**	Enables the debugging of the DPE connection, which logs communication subsystem status and error messages. Use this command to identify communication problems between the DPE and the RDU. To disable debugging of the DPE connection, use the **no** form of this command.	

	Examples
	
	This result occurs when you enable debugging of the DPE connection.
	
	`bac_dpe# debug dpe connection`
	`% OK`
	
	This result occurs when you disable debugging of the DPE connection.
	
	`bac_dpe# no debug dpe connection`
	`% OK`
		Defaults
		
		Debugging of the DPE connection is by default disabled.

debug dpe dpe-server **no debug dpe dpe-server**	Enables debugging of the DPE server, which involves logging messages about the overall status and issues of the DPE server. To disable the debugging of the DPE server, use the **no** form of this command.	

	Examples
	
	This result occurs when you enable debugging of the DPE server.
	
	`bac_dpe# debug dpe dpe-server`
	`% OK`
	
	This result occurs when you disable debugging of the DPE server.
	
	`bac_dpe# no debug dpe dpe-server`
	`% OK`
		Defaults
		
		Debugging of the DPE server is by default disabled.

debug dpe event-manager **no debug dpe event-manager**	Enables debugging of the DPE event manager, which involves logging messages and conditions showing the state of the event manager. To disable debugging of the DPE event manager, use the **no** form of this command.	

	Examples
	
	This result occurs when you enable debugging of the DPE event manager.
	
	`bac_dpe# debug dpe event-manager`
	`% OK`
	
	This result occurs when you disable debugging of the DPE event manager.
	
	`bac_dpe# no debug dpe event-manager`
	`% OK`
		Defaults
		
		Debugging of the DPE event manager is by default enabled.

Table 6-1 List of debug dpe Commands (continued)

Command	Description	
debug dpe exceptions **no debug dpe exceptions**	Enables the debugging of DPE exceptions, which involves logging full stack traces for exceptions occurring during system operation. In unusual situations, such as when the system is apparently corrupt or behaving abnormally, this command can provide valuable information for Cisco support. To disable the debugging of DPE exceptions, use the **no** form of this command.	

	Examples
	
	This result occurs when you enable debugging of DPE exceptions.
	
	`bac_dpe# debug dpe exceptions`
	`% OK`
	
	This result occurs when you disable debugging of DPE exceptions.
	
	`bac_dpe# no debug dpe exceptions`
	`% OK`
		Defaults
		
		Debugging of DPE exceptions is by default enabled.

debug dpe framework **no debug dpe framework**	Enables the debugging of the DPE framework, which involves logging information about the underlying framework of the DPE server. This infrastructure provides for all the various servers in Prime Cable Provisioning. To disable the debugging of the DPE framework, use the **no** form of this command.	

	Examples
	
	This result occurs when you enable debugging of the DPE framework.
	
	`bac_dpe# debug dpe framework`
	`% OK`
	
	This result occurs when you disable debugging of the DPE framework.
	
	`bac_dpe# no debug dpe framework`
	`% OK`
		Defaults
		
		Debugging of the DPE framework is by default enabled.

Table 6-1 List of debug dpe Commands (continued)

Command	Description	
debug dpe messaging **no debug dpe messaging**	Enables debugging of DPE messaging, which involves logging details about the DPE messaging subsystem. This subsystem is used primarily for communication between the DPE and the RDU. To disable the debugging of DPE messaging, use the **no** form of this command.	

	Examples
	
	This result occurs when you enable debugging of DPE messaging.
	
	`bac_dpe# debug dpe messaging`
	`% OK`
	
	This result occurs when you disable debugging of DPE messaging.
	
	`bac_dpe# no debug dpe messaging`
	`% OK`
		Defaults
		
		Debugging of DPE messaging is by default disabled.

debug dpe ssl_all **no debug dpe ssl_all**	Enables the detailed JSSE internal messaging category for debugging SSL messages To disable the internal debugging of JSSE, use the no form of this command.	

	Examples
	
	This result occurs when you enable debugging of JSSE internal messaging category.
	
	`bac_dpe# debug dpe ssl_all`
	`% OK`
	
	This result occurs when you disable debugging of JSSE internal messaging category.
	
	`bac_dpe# no debug dpe ssl_all`
	`% OK`
		Defaults
		
		Debugging of JSSE internal messaging is by default disabled.

debug dpe secure_messaging **no debug dpe secure_messaging**	Enables the basic SSL messaging category for debug messages. To disable SSL messaging category for debug messages.	

	Examples
	
	This result occurs when you enable debugging of basic SSL connections.
	
	`bac_dpe# debug dpe secure_messaging`
	`% OK`
	
	This result occurs when you disable debugging of basic SSL connections.
	
	`bac_dpe# no debug dpe secure_messaging`
	`% OK`
		Defaults
		
		Debugging of basic SSL connections is by default disabled.

debug on

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

Caution Enabling debug logging may have a severe impact on DPE performance. Do not leave the DPE running with debug turned on for an extended period of time.

To disable all the categories of debug logging, run the **no debug all** command. See no debug all, page 6-9.

Syntax Description

No keywords or arguments.

Defaults

Debugging is by default disabled.

Examples

```
bac_dpe# debug on
% OK
```

debug service tftp ipv4 | ipv6

Use the **debug service tftp ipv4 | ipv6** command to enable debugging of TFTP transfers for IPv4 or IPv6.

To disable debugging of the TFTP service, use the **no** form of this command. See no debug service tftp ipv4 | ipv6, page 6-8.

Syntax Description

debug service tftp *1* **ipv4 | ipv6**

- *1*—Identifies the instance of the TFTP service on the DPE.
- **ipv4**—Specifies debugging of the TFTP service for IPv4.
- **ipv6**—Specifies debugging of the TFTP service for IPv6.

Defaults

Debugging of the TFTP service is by default disabled.

Examples

This result occurs when you enable debugging of the TFTP service for IPv4.

```
bac_dpe# debug service tftp 1 ipv4
% OK
```

This result occurs when you enable debugging of the TFTP service for IPv6.

```
bac_dpe# debug service tftp 1 ipv6
```

```
% OK
```

no debug service tftp ipv4 | ipv6

Use the **no debug service tftp ipv4 | ipv6** command to disable debugging of TFTP transfers for IPv4 or IPv6.

To enable debugging of the TFTP service, see debug service tftp ipv4 | ipv6, page 6-7.

Syntax Description

no debug service tftp *1* **ipv4 | ipv6**

- *1*—Identifies the instance of the TFTP service on the DPE.
- **ipv4**—Specifies debugging of the TFTP service for IPv4.
- **ipv6**—Specifies debugging of the TFTP service for IPv6.

Defaults

Debugging of the TFTP service is by default disabled.

Examples This result occurs when you disable debugging of the TFTP service for IPv4.

```
bac_dpe# no debug service tftp 1 ipv4
% OK
```

This result occurs when you disable debugging of the TFTP service for IPv6.

```
bac_dpe# no debug service tftp 1 ipv6
% OK
```

no debug all

Use the **no debug all** command to disable all the categories of debug logging.

For details about enabling debug logging, see debug on, page 6-7.

Syntax Description No keywords or arguments.

Defaults Debug logging is by default disabled.

Examples

```
bac_dpe# no debug all
% OK
```

log level

Use the **log level** command to set the level of minimum DPE log messages that are saved, as described in the *Cisco Prime Cable Provisioning 6.1.3 User Guide*.

Syntax Description **log level** *number*

number—Identifies the logging level, by number, to be saved. Table 6-2 describes the log levels that Prime Cable Provisioning supports.

Table 6-2 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.

Table 6-2 DPE Log Levels (continued)

Log Level No.	Description
5-notification	Saves all notification messages and those of a more severe nature.
6-info	Saves all logging messages available.

Note 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 default log level is 5-notification.

Examples

```
bac_dpe# log level 6
% OK
```

show log

Use the **show log** command to show all recent log entries for the DPE. These logs contain general DPE process information, including 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. See debug dpe, page 6-3, for detailed information.

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 keyword 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 keyword is optional.

Defaults No default behavior or values.

Examples This result occurs when you use the **show log** command.

```
bac_dpe# show log
dpe.example.com: 2007 06 04 08:01:42 EDT: %BPR-DPE-5-0236: [Device Provisioning Engine]
starting up.
```

```
dpe.example.com: 2007 06 04 08:01:42 EDT: %BPR-DPE-6-0822: Server version [Cisco Prime
Cable Provisioning 5.1 (SOL_BAC5_1_0_00000000_0505)].
dpe.example.com: 2007 06 04 08:01:42 EDT: %BPR-DPE-6-0689: Maximum Java heap size [307
MiB].
dpe.example.com: 2007 06 04 08:01:42 EDT: %BPR-DPE-6-0690: Maximum database cache size
[102 MiB].
dpe.example.com: 2007 06 04 08:01:42 EDT: %BPR-DPE-5-1360: Connecting to RDU
[dpe.example.com:49187]. Rate [1/d].
dpe.example.com: 2007 06 04 08:05:31 EDT: %BPR-DPE-5-0195: Connected to RDU
[dpe.example.com:49187]. Time to connect [3.8 min]. Rate [1/d].
dpe.example.com: 2007 06 04 08:05:31 EDT: %BPR-DPE-5-0982: Configured provisioning
interfaces: [localhost[10.10.0.1]].
dpe.example.com: 2007 06 04 08:05:31 EDT: %BPR-DPE-5-1359: Batch
[DPE:dpe.example.com/10.86.149.133:bf7190:112f6a01cf7:80000002]. Registering with RDU.
Rate [1/d].
dpe.example.com: 2007 06 04 08:05:32 EDT: %BPR-LICENSING-3-0998: Server registration
failed. Lack of DPE licenses.
dpe.example.com: 2007 06 04 08:05:33 EDT: %BPR-DPE-5-1374: Opening database [default.db].
dpe.example.com: 2007 06 04 08:05:34 EDT: %BPR-DPE-5-1375: Opened database [default.db].
Time to open [1.2 s].
dpe.example.com: 2007 06 04 08:05:34 EDT: %BPR-TFTP-5-0462: Service is disabled.
dpe.example.com: 2007 06 04 08:05:34 EDT: %BPR-TOD-5-5501: TOD Server disabled.
dpe.example.com: 2007 06 04 08:19:21 EDT: %BPR-LICENSING-5-1002: DPE received a license
event from the RDU.
dpe.example.com: 2006 12 21 11:22:20 GMT: %BPR-DPE-5: DPE-0: Device Provisioning Engine
starting up
...
```

Note The output presented in this example is trimmed for demonstration purposes.

This result occurs when you use the **show log last** command.

```
bac_dpe# show log last 2
dpe.example.com: 2007 06 04 08:19:23 EDT: %BPR-DPE-5-0147: Batch dpe.example.com: 2007 06
04 08:19:23 EDT: %BPR-DPE-5-1371: Synchronized [0] cached device configurations with RDU.
Time to synchronize [52 ms] ([0/s]).
dpe.example.com: 2006 12 21 11:28:17 GMT: %BPR-DPE-5: DPE-0: Device Provisioning Engine
starting up
```

This result occurs when you use the **show log run** command.

```
dpe# show log run
Press <enter> to stop.
dpe.example.com: 2006 12 21 11:43:43 GMT: %BPR-DPE-5: DPE-0: Device Provisioning Engine
starting up
dpe.example.com: 2006 12 21 11:43:44 GMT: %BPR-DPE-5: Info DPE: Attempt to connect to RDU
BPR_host.example.com:49187 failed;
dpe.example.com: 2006 12 21 11:43:44 GMT: %BPR-DPE-5: Info TFTP: Ready to service requests

% Stopped.
```

Support and Troubleshooting Commands

This chapter contains the command-line interface (CLI) commands that you can use to support troubleshooting for the Prime Cable Provisioning Device Provisioning Engine (DPE).

The commands described in this chapter include:

Command	Description	CLI Mode		Required Privileges			
		Login	Enable	PRIV_DPE_READ	PRIV_DPE_UPDATE	PRIV_DPE_SECURITY	PRIV_DEVICE_READ
clear bundles	Clears existing archived bundles on the DPE.		✓	✓	✓		
show bundles	Displays bundles currently available in the outgoing directory.	✓	✓	✓			
support bundle cache	Bundles the current DPE cache.		✓	✓	✓		

clear bundles

Use the **clear bundles** command to clear existing archived bundles on the DPE. These bundles, which you create using the **support bundle cache** command, normally contain archived logs and archived state information, which are of use to the Cisco Technical Assistance Center.

Caution Before using the **clear bundles** command, ensure that you retrieve all bundles because you will lose the archived state.

Once you enter this command, a prompt appears to indicate that the bundles are being cleared. When bundling is complete, the amount of disk space cleared (in bytes) appears.

Syntax Description No keywords or arguments.

Defaults No default behavior or values.

Examples This result occurs when existing archived bundles are cleared.

```
bac_dpe# clear bundles
Clearing Cisco support bundles...
+ 89088 bytes cleared.
```

This result occurs when there are no archived bundles to clear.

```
bac_dpe# clear bundles
Clearing Cisco support bundles...
+ No bundles to clear.
```

show bundles

Use the **show bundles** command to display the bundles currently available in the outgoing directory. The bundles, which you create using the **support bundle cache** command, are accessible from the FTP server of the DPE.

This command identifies the bundles that are archived. If there are no bundles, a prompt appears indicating that no bundles are available.

Syntax Description No keywords or arguments.

Defaults No default behavior or values.

Examples This result occurs when bundles are archived.

```
bac_dpe# show bundles
outgoing/state-20070608-043109.bpr
outgoing/cache-20070608-043150.bpr
```

This result occurs when there are no archived bundles.

```
bac_dpe# show bundles
No bundles currently available.
```

support bundle cache

Use the **support bundle cache** command to bundle the current DPE cache. This command is useful when archiving the cache for delivery to the Cisco Technical Assistance Center. Once the bundle is created, it is available from the outgoing directory of the FTP server.

After the command creates the cache bundle, it displays the bundle specifics, including the compressed size of the bundle file.

Syntax Description No keywords or arguments.

Defaults No default behavior or values.

Examples

```
bac_dpe# support bundle cache
Creating cache bundle for Cisco support...
+ outgoing/cache-20071008-070730.bpr
+ Adding & compressing DPE cache...
+ Size: 23155 bytes
```

Event System Management Commands

This chapter describes the command-line interface (CLI) commands used the PCP DPE to trigger event and, monitor and manage the Prime Cable Provisioning event system.

For information on **DPE Event Publisher**, see *Cisco Prime Cable Provisioning 6.1.3 User Guide*.

Note Before using a DPE event command, you must enable DPE event monitor by running the DPE event monitor command.

If you run the following commands on an unlicensed DPE, the following message appears:

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

The commands described in this chapter are:

Command	Description	CLI Mode		Required Privileges			
		Enable	Disable	PRIV_DPE_READ	PRIV_DPE_UPDATE	PRIV_DPE_SECURITY	PRIV_DEVICE_READ
dpe event monitor	Enable the DPE event monitor	✓		✓	✓		
dpe event request	Events the DPE request service	✓		✓	✓		
dpe event config	Events the DPE cache configuration	✓		✓	✓		
dpe event file	Events the DPE cache file operation	✓		✓	✓		
dpe event log	Events the DPE logs with log level	✓		✓	✓		
dpe event tftp	Events the DPE TFTP requests	✓		✓	✓		

Command	Description	CLI Mode		Required Privileges			
		Enable	Disable	PRIV_DPE_READ	PRIV_DPE_UPDATE	PRIV_DPE_SECURITY	PRIV_DEVICE_READ
no dpe event monitor	Disable all the DPE events		✓	✓	✓		
no dpe event request	Stop Events the DPE request service		✓	✓	✓		
no dpe event config	Stop Events the DPE cache configuration		✓	✓	✓		
no dpe event file	Stop Events the DPE cache file operation		✓	✓	✓		
no dpe event log	Stop Events the DPE logs with log level		✓	✓	✓		
no dpe event tftp	Stop Events the DPE TFTP requests		✓	✓	✓		

dpe event

Use the dpe event command to configure event settings on the DPE. Following are the examples that you can use with this command:

Syntax Description

dpe event monitor

no dpe event monitor

Enables DPE to trigger events, which involves common interface to enable DPE events features and depends on the below event type. It triggers the DPE events.

- *dpe event config*—Enables dpe configuration event for DPE event.
- *dpe event file*—Enables dpe file event for DPE event.
- *dpe event log*—Enables dpe log level event for DPE event.
- *dpe event request*—Enables dpe request event for DPE event.
- *dpe event tftp*—Enables dpe tftp event for DPE event.

To disable all the DPE events, use the no form of this command.

Examples

This result occurs when you enable dpe event monitor of the DPE.

```
bac_dpe# dpe event monitor
% OK
```

This result occurs when you disable dpe event monitor of the DPE.

```
bac_dpe# no dpe event monitor
% OK
```

Defaults Event of the DPE is by default disabled.

dpe event config

Syntax Description **dpe event config**

no dpe event config

Enables configuration events to the DPE, Which involves events that occurs while configuration changes in the dpe cache. It includes,

- Received configuration for device from RDU.
- Received updated configuration for device from RDU.
- Removed configuration for device from cache.
- Completed device attributes dumping process.

To disable DPE configuration events, use the no form of this command.

Examples This result occurs when you enable configuration event of the DPE.

```
bac_dpe# dpe event config
% OK
```

This result occurs when you disable configuration event of the DPE.

```
bac_dpe# no dpe event config
% OK
```

Defaults Configuration event of the DPE is by default disabled.

dpe event file

Syntax Description **dpe event file**

no dpe event file

Enables file events to the DPE, Which involves events that occurs while file configuration changes in the DPE cache. It includes,

- Received file from RDU.
- Received updated file from RDU.
- Removed file from cache.

To disable DPE file events, use the no form of this command.

Examples

This result occurs when you enable file event of the DPE.

```
bac_dpe# dpe event file
% OK
```

This result occurs when you disable file event of the DPE.

```
bac_dpe# no dpe event file
% OK
```

Defaults

File event of the DPE is by default disabled.

dpe event request

Syntax Description

dpe event request

no dpe event request

Enables dpe request events to the DPE, Which involves events that occurs while device request, ToD request, SNMP reset request. It includes,

Device request

- Sending no cached configuration for device in provisioning group to device.
- Sending configuration for device in provisioning group to device.

ToD request

- Received UDP time of day request from device.
- ToD Success/Failure.

SNMP reset

- Processing SNMP reset for device.
- Successfully send SNMP reset for device.

To disable DPE request events, use the no form of this command.

Examples This result occurs when you enable dpe request event of the DPE.

```
bac_dpe# dpe event request
% OK
```

This result occurs when you disable dpe request event of the DPE.

```
bac_dpe# no dpe event request
% OK
```

Defaults Request event of the DPE is by default disabled.

dpe event tftp

Syntax Description **dpe event tftp**

no dpe event tftp

Enables tftp events to the DPE, Which involves events that occurs while device request for tftp. It includes,

- Received a TFTP [read] request from device for file.
- Finished handling [read] request from device for file.
- TFTP exception.

To disable DPE tftp events, use the no form of this command.

Examples This result occurs when you enable dpe tftp event of the DPE.

```
bac_dpe# dpe event tftp
% OK
```

This result occurs when you disable dpe tftp event of the DPE.

```
bac_dpe# no dpe event tftp
% OK
```

Defaults TFTP event of the DPE is by default disabled.

dpe event log

Syntax Description

dpe event log

no dpe event log

Enables log events to the DPE, Which involves events that occurs for all dpe process and send the log event depends on the log level of DPE. It includes,

- Send the DPE log as events.
- Depend on the DPE log level it send the logs as events.

To disable DPE log events, use the no form of this command.

Examples

This result occurs when you enable dpe log event of the DPE.

```
bac_dpe# dpe event log
% OK
```

This result occurs when you disable dpe log event of the DPE.

```
bac_dpe# no dpe event log
% OK
```

Defaults

Log event of the DPE is by default disabled.

A

alert A syslog or SNMP message notifying an operator or administrator of a network problem.

API Application programming interface. Specification of function-call conventions that defines an interface to a service.

B

Cisco Prime Cable Provisioning An integrated solution for data-over-cable service providers to configure and manage broadband modems, and enable and administer subscriber self-registration and activation.
Prime Cable Provisioning is a scalable product capable of supporting millions of devices.

bandwidth The difference between the highest and lowest frequencies available for network signals. Also used to describe the rated throughput capacity of a given network medium or protocol.

broadband A transmission system that multiplexes multiple independent signals onto one cable. In telecommunications terminology, any channel having a bandwidth greater than a voice-grade channel (4 kHz); in LAN terminology, a coaxial cable on which analog signaling is used.

Prime Cable Provisioning *See* Cisco Prime Cable Provisioning.

C

cable modem termination system *See* CMTS.

CableHome A CableLabs initiative to develop a standardized infrastructure to let cable operators extend high-quality, value-added services to the home local-area network.

caching A form of replication in which information learned during a previous transaction is used to process later transactions.

CMTS Cable modem termination system. A component that exchanges digital signals with cable modems on a cable network. The CMTS is usually located in the local office of the cable provider.

CMTS shared secret *See* shared secret.

configuration file A file containing configuration parameters for the device to be provisioned.

CPE Customer premises equipment. Terminating equipment, such as telephones, computers, and modems, that are supplied and installed at a customer location.

D

DOCSIS Data Over Cable Service Interface Specification. Defines functionality in cable modems involved in high-speed data distribution over cable television system networks.

DPE Device Provisioning Engine. Distributed servers that cache device information and that automatically synchronize with the RDU to obtain the latest configurations and provide Prime Cable Provisioning scalability.

F

FQDN Fully qualified domain name. The full name of a system, rather than just its hostname; for example, cisco is a hostname and www.cisco.com is an FQDN.

I

Internet Protocol (IP, IPv4) Network layer for the TCP/IP protocol suite. Internet Protocol (version 4) is a connectionless, best-effort packet switching protocol. Defined in RFC 791.

IP address A 32-bit number assigned to hosts using TCP/IP that identifies each sender or receiver of information that is sent in packets across the Internet.

IPv6 IP version 6. Replacement for the current version of IP (version 4). IPv6 includes support for flow ID in the packet header, which can be used to identify flows. Formerly called IPng (next generation).

K

KDC Key Distribution Center. Implements limited Kerberos functionality and is used in the provisioning of PacketCable MTAs.

M

MAC address Standardized data-link layer address that is required for every port or device that connects to a LAN. Other devices in the network use these addresses to locate specific ports in the network and to create and update routing tables and data structures. MAC addresses are 6 bytes long and are controlled by IEEE, Also known as hardware address, MAC-layer address, or physical address.

Media Terminal Adapter *See* MTA.

MSO Multiple system operator. A company that operates more than one cable TV or broadband system.

MTA Equipment at the customer end of a broadband (PacketCable) network.

multiple service operator *See* MSO.

N

NAT Network address translation. Mechanism for reducing the need for globally unique IP addresses. NAT allows an organization with addresses that are not globally unique to connect to the Internet by translating those addresses into globally routable address space. Also known as Network Address Translation.

network administrator Person responsible for operation, maintenance, and management of a network. *See also* network operator.

network operator Person who routinely monitors and controls a network, performing such tasks as reviewing and responding to alarms, monitoring throughput, configuring new circuits, and resolving problems. *See also* network administrator.

Network Time Protocol *See* NTP.

NR Cisco Network Registrar. A software product that provides IP addresses, configuration parameters, and DNS names to DOCSIS cable modems and PCs, based on network and service policies.

NTP Network Time Protocol. A protocol designed to synchronize server clocks over a network.

P

PacketCable A CableLabs initiative for interoperable interface specifications to deliver advanced, real-time multimedia services over a two-way cable network. Built on top of cable modem infrastructure to enable a wide range of multimedia services, such as IP telephony, multimedia conferencing, interactive gaming, and general multimedia applications.

provisioning API A series of Prime Cable Provisioning functions that programs can use to make the operating system perform various functions.

provisioning groups Groupings of devices with a defined set of associated DPE and DHCP servers, based on either network topology or geography.

R

RDU Regional Distribution Unit. The primary server in the Prime Cable Provisioning provisioning system, manages generation of device configurations, processes all API requests, and manages the Prime Cable Provisioning system.

realm Logical network served by a single Kerberos database and a set of Key Distribution Centers.

realm names By convention, realm names are all uppercase letters to differentiate the realm from the Internet domain. *See* realm.

redundancy In internetworking, the duplication of devices, services, or connections so that, in the event of a failure, the redundant devices, services, or connections can perform the work of those that failed.

S

selection tags Selection tags associated with Network Registrar scopes. Define the clients and client classes associated with a scope.

shared secret A character string used to provide secure communication between two servers or devices.

T

TFTP Trivial File Transfer Protocol. Simplified version of File Transfer Protocol (FTP) that allows files to be transferred from one computer to another over a network.

W

watchdog A daemon process used to monitor, stop, start, and restart Prime Cable Provisioning component processes such as the RDU, Tomcat, and the SNMP agent.

INDEX

A

B

C

D

E

F

G

H

I

K

L

M

P

R

S

T

U

V