



Preparing for the Installation

This chapter describes the steps required to set up your Oracle Database 11g server and to set up the Cisco Configuration Engine with Prime Provisioning.

- [Setting Up Oracle for Prime Provisioning, page 2-1](#)
- [Setting up Cisco Configuration Engine with Prime Provisioning, page 2-9](#)

Setting Up Oracle for Prime Provisioning

Prime Provisioning comes with an embedded Sybase DBMS. This section is relevant to you only if you choose to use an external Oracle DBMS. You are responsible for the installation, licensing, and administration of the Oracle DBMS.

This section describes how to set up an Oracle Database 11g, Enterprise Edition Release 11.2.0.3.0 - 64 bit Production server that works with Cisco Prime Provisioning. This section is intended for database administrators who are familiar with Oracle.



Note

Prime Provisioning 6.6 was tested with Oracle Database 11g, Enterprise Edition Release 11.2.0.3.0 - 64 bit Production. If you would like to use another version of Oracle, see Oracle's compatibility information.

This chapter does not cover all the details about installing and setting up this Oracle server. For the complete information, see the Oracle Installation Guide. Prime Provisioning provides schema files to be loaded on an Oracle server. The Prime Provisioning customer must decide on the Oracle server configuration.

1. [Prerequisites, page 2-2](#)
2. [Installing Oracle, page 2-2](#)
3. [Verifying and Launching Oracle, page 2-3](#)
4. [Setting Up Your Oracle Files, page 2-4](#)
5. [Testing Your Oracle Database Connection for Oracle User Prime, page 2-5](#)
6. [Prime Provisioning Software Installation, page 2-6](#)
7. [Verify Prime Provisioning Installation with Oracle, page 2-6](#)
8. [Importing an Oracle Repository Dump, page 2-6](#)

9. [Configuring Oracle RAC, page 2-7](#)
10. [Backup of Oracle Database, page 2-8](#)

This section also contains a “[Troubleshooting](#)” section on page 2-8.

Prerequisites

Prime Provisioning support for an Oracle database is for Oracle Database 11g, Enterprise Edition Release 11.2.3.1.0 - 64 bit Production. This is the version of Oracle with which Prime Provisioning 6.6 was tested. If you would like to use another version, see Oracle’s compatibility information.

The remaining prerequisites are as specified in the following steps:

Step 1 When the Oracle server is set up, the following initialization parameters should be in the database **init** file:

- db_block_size = 8192 or larger
- compatible = “11.2.0.3.0”
- open_cursors = 512 or larger
- processes = 150 or larger

Step 2 Record the following information about the server setup. This information is needed during the Prime Provisioning installation:

- Oracle server name
- Oracle server instance identifier (SID)



Note This is specified in [Step 18](#).

- Database port number for client connections (default: 1521)
- Oracle user ID and password created for Prime Provisioning



Note Create an Oracle database userid and password. This is needed during Prime Provisioning installation. Do not use the **system** or **sys** account for Prime Provisioning data. Use a separate table space other than the system table space.

Step 3 Before loading the Prime Provisioning database schema, make sure the Oracle database has been successfully started and the database user has proper privileges. See the Oracle Administration Guide for detailed instructions about how to set up the database and manage user accounts.

Step 4 Proceed to the section “[Installing Oracle](#).”

Installing Oracle

The following information about an Oracle installation is just one example.

You must install Oracle before you install the Cisco PRIME Fulfillment (Prime Provisioning) software (or at least know your Oracle home directory, host machine, and Oracle Server ID), and your database and its listener must be running when you launch the Prime Provisioning servers.

If you intend to use the same Oracle installation with more than one installation of the Prime Provisioning servers, you must create a unique Oracle SID and Oracle tablespace for each Prime Provisioning installation.

initORACLE_SID.ora

This file should already exist in the `/dbs` subdirectory of your Oracle installation. (The filename contains your database's SID in place of `ORACLE_SID`. For example, if you named your database Prime Provisioning, this file is named `initISC.ora`.)

oratab

The `oratab` file should be located in the `/var/opt/oracle` directory on the machine on which the database is installed. It is used by Oracle's **dbstart** utility to identify your database.

The `oratab` file must contain the following line:

```
database_name:location_of_your_Oracle_executables:Y
```

If your Oracle home directory is `/oracle/10.2.0` and your database SID is Prime Provisioning, the `oratab` entry would be as follows:

```
Prime Provisioning:/oracle/10.2.0:Y
```

This file identifies the name and location of your database for the Oracle utility **dbstart** (and its companion **dbshut**). The **dbstart** utility starts Oracle; the "Y" at the end of the `oratab` entry tells the **dbstart** utility to open the database named Prime Provisioning. (Substitute your database name for Prime Provisioning in the sample. List the path to your Oracle installation as an absolute path, not a relative path.)

To make this happen automatically following a reboot (after a power interruption, for example), execute the **dbstart** utility from a script in the `/etc/init.d` directory on the Oracle host machine.

Verifying and Launching Oracle

Your Oracle database must be open before you can install or use the Prime Provisioning software.

First, verify the Oracle processes, as described in the following section. If the processes are running, you can skip the succeeding section.

Verifying Oracle Processes

Log into the Oracle host machine and enter the following on the command line to see if the Oracle processes are running:

```
ps -ef | grep ora_
```

```
ps -ef | grep tnslnr
```

If there is no output displayed from the `ps` command, Oracle is not running.

If Oracle is running and the listener process is running, you should see something similar to the following:

```

oracle 328 1 0 14:25:18 0:00 ora_pmon_ISC
oracle 328 1 0 14:25:18 0:00 ora_dbwr_ISC
oracle 328 1 0 14:25:18 0:00 ora_lgwr_ISC
oracle 328 1 0 14:25:18 0:00 ora_ckpt_ISC
oracle 328 1 0 14:25:18 0:00 ora_smon_ISC
oracle 328 1 0 14:25:18 0:00 ora_reco_ISC
oracle 328 1 0 14:25:18 0:00 ora_wmon_ISC
oracle 328 1 0 14:25:18 0:00 tnslnsr LISTENER -inherit

```

These are the Oracle processes currently running (your output might not match this list exactly, depending on which Oracle components are installed).

Launching Oracle and Opening Your Database

Your Oracle database must be open before you can install or use the Prime Provisioning software.

If Oracle is not currently running, you must use the startup utilities located in the `/bin` subdirectory of your Oracle installation.

To open your database, you must be logged into the Oracle host workstation under the Oracle administrator (DBA) user ID; you then locate your `$ORACLE_HOME/bin` subdirectory.

On the command line, enter the following:

dbstart

The `dbstart` script starts the database identified in the `oratab` file. If the database starts successfully, you should see several lines of output, including the following:

```

SQL> Connected to an idle instance.
SQL> ORACLE instance started.

```

...and ending with the following:

```

Server Manager Complete.
Database "Prime Provisioning" warm started.

```

If the listener process is not running, you must also start that process. On the command line, enter the following:

lsnrctl start

You should see several lines of output as the process is invoked, then you should see output similar to the following:

```

Services Summary...
  Prime Provisioning      has 1 Service handler(s)

```

The command completed successfully.

Setting Up Your Oracle Files

To configure your database to work with the Prime Provisioning software, you must create a tablespace and configure several files.

You must be logged into the Oracle host using the user ID (such as `oracle`) created during the Oracle installation procedure.

Oracle Tablespace Requirements

You must create an Oracle tablespace for your Prime Provisioning tables.

To create the tablespace, Oracle must be running and your database must be open.

Log into the Oracle host using the `oracle` user ID. Identify (or create) the directory where your Prime Provisioning data should be stored, and grant write permission to the `oracle` user ID. Be sure your `ORACLE_SID` and `ORACLE_HOME` environment variables are set correctly, then launch the Oracle utility `sqlplus`, which is located in the `$ORACLE_HOME/bin` directory.

At the SQL prompt, enter the following on the command line:

```
connect / as sysdba;
```

```
CREATE TABLESPACE ISC_DAT
```

```
DATAFILE 'your_data_directory/ISC_DAT_01.dbf' size 500M
```

```
autoextend on
```

```
next 50M
```

```
maxsize unlimited;
```

The data directory you specify must already exist. The `TABLESPACE` and `DATAFILE` names are arbitrary. You can use any names that help you keep track of which files are associated with which database. The only requirement is that the name given to the tablespace at the time of its creation (`ISC_DAT` in the example) must be the same as the default tablespace listed when you create the `prime` user account.

The autoextend option allows ORACLE to automatically extend your data file. The maximum size of the data file is limited only by the available space on the file's disk.

prime Oracle User Account

Before invoking an Oracle installation, you need to grant 'create and view' permissions to an Oracle user. To grant these permissions, execute the SQL query as shown in the following steps:

While `sqlplus` is still running, create a `prime` user account using your `ISC_DAT` tablespace as follows:

```
CREATE USER prime IDENTIFIED BY cisco
```

```
DEFAULT TABLESPACE ISC_DAT;
```

```
GRANT CONNECT TO prime;
```

```
GRANT RESOURCE TO prime;
```

```
GRANT CREATE VIEW TO <<PRIME PROVISIONING DB username>>
```

```
GRANT SELECT ON sys.dba_constraints TO <<PRIME PROVISIONING Oracle DB username>>
```

You should use the username and password created in these steps when entering Oracle information in the script `prime.configure`.

Testing Your Oracle Database Connection for Oracle User Prime

When you have configured your database and listener file, enter the following (for the Oracle user `prime` and for the database named Prime Provisioning) on the command line:

```
sqlplus <username>/<password>
```

<username> is a database username (in our previous example, we used `prime`).

`<password>` is a database password (in our previous example, we used **cisco**).

If your system is set up properly (and your Oracle database is running), you should see a message advising you that you are connected to Oracle. Enter `quit` on the command line to exit the database.

Prime Provisioning Software Installation



Note

The Prime Provisioning database schema files are loaded during the installation.

Perform the following:

- Step 1** Follow the **custom** install instructions in “Installing Prime Provisioning” section on page 3-2 , and log in, as explained in the “Logging In for the First Time” section on page 5-3.
- Step 2** Proceed to the section “Verify Prime Provisioning Installation with Oracle”.

Verify Prime Provisioning Installation with Oracle

To verify the Prime Provisioning installation with Oracle, do the following:

- Step 1** Run **sqlplus** `<oracle_id>/<oracle_password>` on the Oracle server.
- Step 2** From the **SQL>** prompt, run **select host_name from vpnc_host;**
This command returns the installed Prime Provisioning hostname.
- Step 3** Log into the Prime Provisioning server.
- Step 4** Check the file `/opt/PrimeProvisioning/etc/vpnc.properties` and make sure that the `<oracle server>` and `<ORACLE_SID>` are correct in the following entry in the file:
`repository.persistence.url=jdbc:oracle:thin:@<oracle server>:<ORACLE_SID>`
- Step 5** Execute the schema verification script to verify the repository schema version, as follows:
`cd $PRIMEP_HOME`
`cd /bin`
`./checkSchemaVer.sh <oracle_id>/<oracle_password>`
where: `<oracle_id>` is the Prime Provisioning userid in the Oracle database and `<oracle_password>` is its password.
- Step 6** The output from the script should be “Current schema version = 6.0”. If that is not the output from the script, Prime Provisioning might not have been installed properly or the Prime Provisioning repository might not have been upgraded successfully.

Importing an Oracle Repository Dump

To import the Oracle repository dump in Prime Provisioning, do the following:

-
- Step 1** Log into the Oracle Webapp.
- Step 2** Create the User and Tablespace.
- Step 3** Log into the Oracle Server and source the environment:
- ```
rlogin <Oracle Server Name>
```
- where:
- <Oracle Server Name> Specify the Oracle Server Name that is being used.
- Step 4** Enter: `su - <user name>`
- Step 5** Enter: `cd $ORACLE_HOME/bin`
- Step 6** Enter: `source coraenv`
- Step 7** Enter: `setenv ORACLE_HOME to $ORACLE_HOME/bin`
- Step 8** Enter: `setenv ORACLE_SID to orcl`
- Step 9** Copy the .dmp file to a directory on the Oracle Server.
- Step 10** Enter: `cd $ORACLE_HOME/bin`
- Step 11** Run the command: `imp`
- When you run this script, you are asked to enter the values for the following prompts:
- Import file: <specify the full path to .dmp file>
  - Enter insert buffer size: **30720** (accept the default value)
  - List contents of import file only: **no** (accept the default value)
  - Ignore create error due to object existence: **no** (accept the default value)
  - Import grants: **no** (accept the default value)
  - Import table data: **yes**
  - Import entire export file: **yes**
- 

## Configuring Oracle RAC

In addition to having already installed Prime Provisioning and followed the steps required to configure an Oracle server, you must follow these steps when using Oracle Real Application Clusters (RAC). Prime Provisioning does not support client load balancing with Oracle RAC.



### Note

A limitation of Oracle RAC is that any uncommitted transactions made during an instance or node failure and recovery period are lost. The recovery of these transactions is not supported. For this reason, the behavior of tasks that are running at the time as an instance or node fail over is undetermined. These tasks should be redeployed.

In case of a failure, for more information see the Oracle RAC documentation for database instance recovery time details.

---

- 
- Step 1** Verify that the new Oracle RAC servers are available and have an Prime Provisioning tablespace with user configured. If you need help setting this up, see the [“Verify Prime Provisioning Installation with Oracle”](#) section on page 6.
- Step 2** Modify \$PRIMEP\_HOME/runtime.properties to have the correct values for the following parameters:
- **db\_server**
  - **db\_url**—A **sample URL** is jdbc:oracle:thin:@//Virtual IP:<port>/globalSID, where <port> is the port number, which defaults to **1521**.
  - **db\_driver**
  - **db\_usr**
  - **db\_pwd**
- Step 3** Prepopulate the database user name and password into the database  
`./pime.sh execjava.sh com.cisco.vpnsc.common.BootStrapHelper put repository <oracle username> <oracle password>`
- Step 4** If running, use the `./prime.sh stop` command to stop Prime Provisioning.
- Step 5** Verify that the value for the DCPL property watchdog/server/dbpoller/connectionextend is still set to the default: 5. See Appendix C, “DCPL Properties,” in the [Cisco Prime Provisioning 6.6 User Guide](#).
- Step 6** To update the database with the changes, enter:
- ```
./prime.sh startdb
./prime.sh initdb.sh
```
- Step 7** Use `./prime.sh stop` to stop the database.
- Step 8** Then enter `./prime.sh start` to start Prime Provisioning.
-

Backup of Oracle Database

See the [Backup and Restore of Prime Provisioning Repository](#) chapter in the Administration Guide.

Troubleshooting

This section lists Oracle database-related trouble shooting tips based on the following error messages:

- **ORA-01631: max # extents (4096) reached in table xyz**
 If you receive this message, it is typically an Oracle server storage configuration issue. This problem occurs when the tablespace for Prime Provisioning exceeds the limit set by the database configuration. To prevent this, plan proper storage before Prime Provisioning is set up. If this problem occurs, increase the initial or next extent, increase the growth percentage (such as, PCT_INCREASE), or reset the number of max extents (can be unlimited). The Prime Provisioning data must be exported and imported to the tablespace with the new tablespace parameters.
- **Unable to contact Rbac Manager**
 If you receive this message on Prime Provisioning and are unable to log in, this might be because Prime Provisioning cannot connect to the Oracle database. To avoid this situation, increase the number of Oracle server processes.
- **Cannot log into Inventory Manager or Topology Manager**

If you cannot log into the Inventory Manager or Topology Manager, verify that the Oracle hostname is accessible from a client machine, either by DNS or a host file.

- **Resynchronize Prime Provisioning with new or updated Oracle ID and password**

If the Oracle ID and password change after the Prime Provisioning installation, you must execute the following:

- a. `execjava.sh com.cisco.vpnsc.common.BootStrapHelper put repository <oracle_id>
<oracle_password>`
- b. update `etc/spe/cns.properties` and modify these two properties:
`DataAccess.principal.1 <oracle_id>`
`DataAccess.credentials.1 <oracle_password>`

Setting up Cisco Configuration Engine with Prime Provisioning

Overview

This section gives information about downloading to a server using Cisco Configuration Engine with Prime Provisioning.

For versions 2.0, 3.0, and 3.5 of the Cisco Configuration Engine software, the server is a server. For version 1.3.x, 1.4, and 1.5 of the Cisco Configuration Engine software, the server is the Cisco CNS Intelligence Engine 2100 (IE2100) appliance.

Prime Provisioning supports the Device Access Protocol (DAP) of CNS for communication with any Cisco IOS device. The DAP includes:

- uploading a configuration file from a device
- downloading a configlet to a device
- executing a command on a device and obtaining the result (all communications).

Prime Provisioning supports CNS Plug-and-Play.

- [Set Up Steps, page 2-9](#)
- [Checking Router Configurations Overview, page 2-13](#)

Set Up Steps

To enable a server running the Cisco Configuration Engine functionality on Prime Provisioning, set up in the following order:

1. Set up the servers for Cisco Configuration Engine, as shown in “[Set Up to Download to a Server Using Cisco Configuration Engine.](#)”
2. Configure a TIBCO Rendezvous Routing Daemon (**rvrtd**), as shown in “[Configure a TIBCO Rendezvous Routing Daemon.](#)”

Set Up to Download to a Server Using Cisco Configuration Engine

Prime Provisioning supports the integration with servers running the Cisco Configuration Engine 1.3.x, 1.4, 1.5, 2.0, 3.0, and 3.5 software.

For the Cisco Configuration Engine 1.3.x, 1.4, 1.5, 2.0, 3.0, and 3.5 software installation and setup, see the Cisco Configuration Engine 1.3.x documentation set at:

http://www.cisco.com/en/US/products/sw/netmgmtsw/ps4617/tsd_products_support_series_home.html

On a freshly set up Cisco Configuration Engine server, remove Pluto protection, as follows.

-
- Step 1** Log in as **root**.
- Step 2** Enter:
plutosetup.
- Step 3** A warning appears:
“plutosetup will open some class files to public access. It is a security risk.”
Continue (y/n):
Answer **y** for yes to the above warning.



Note

Because the Cisco Configuration Engine server and the Prime Provisioning Master server are behind a secure barrier, we can safely answer **y** for yes to the security risk warning message above. This removal of Pluto protection exposes some files in the Cisco Configuration Engine server that allow Prime Provisioning to create, delete, and edit servers in the Cisco Configuration Engine repository. This is needed for proper Prime Provisioning to Cisco Configuration Engine 1.3.x, 1.4, 1.5, 2.0, 3.0, and 3.5 integration. Removal of Pluto protection only needs to occur when a particular Cisco Configuration Engine server is first used and every time the file **/opt/CSCOcsie/bin/pluto** is deleted for any reason.

Configure a TIBCO Rendezvous Routing Daemon

In this section, do the following:

1. [Configuring the rvrD Daemon on the Prime Provisioning Master Machine, page 2-10.](#)
2. [Configuring the rvrD Daemon on a Cisco Configuration Engine Server, page 2-11.](#)

Configuring the rvrD Daemon on the Prime Provisioning Master Machine

The TIBCO Rendezvous Routing Daemon (**rvrd**) is the default daemon on the Prime Provisioning Master server.

To configure an **rvrd** daemon on an Prime Provisioning Master server:

-
- Step 1** Go to **Administration > Hosts**.
The Hosts page displays the list of hostnames available.
- Step 2** Select your hostname and click **Config**.
The Host Configuration page is displayed.

- Step 3** Go to **CNS > tibco**.
- Step 4** Set the Tibco properties as follows:
- **ConnectInterval** value- 60
 - **logLevel** value - INFO.
 - **Hostname New Value** - 7500.
 - **Network** - enter the local network name created in CNS Server. For example: prime.
 - **Service** - enter the value for the service created in CNS Server. For example: 7530.

Configuring the rvrld Daemon on a Cisco Configuration Engine Server

To configure an **rvrld** daemon on a Cisco Configuration Engine server, do the following:

- Step 1** The TIBCO Rendezvous Routing Daemon (**rvrld**) is the default daemon on the Cisco Configuration Engine server.
- To configure an **rvrld** daemon on a Cisco Configuration Engine server, start a Prime Provisioning-supported browser and go to the following URL:
http://<ciscoconfigurationengine_hostname>:7580 or
http://<ciscoconfigurationengine_ip_address>:7580.
- Step 2** Look at the **component** field under the **information** link to verify that **rvrld** is running. It should say **rvrld**, as shown in [Figure 2-1](#).

Figure 2-1 Cisco Configuration Engine rvrld Verification

The screenshot shows the TIB/Rendezvous web interface. The title is "TIB/Rendezvous" with a sub-header "Routing Daemon - 6.4.8". The URL is "[en2110-1.cisco.com]" and the timestamp is "2003-03-28 17:50:11". On the left, there is a navigation menu with links: "information", "services", "clients", "configure:", "security", "routers", "logging", "copyright", and "web home". The "information" link is selected, and the "Component Information" table is displayed:

Component Information	
component:	rvrld
version:	6.4.8
license ticket:	65598
host name:	en2110-1.cisco.com
user name:	root
IP address:	192.168.116.41
client port:	7500
network services:	5
routing names:	1

- Step 3** Click on the **routers** link in the left column.
- Step 4** In the **Add Router Name** field in the upper part of the window, enter the name of the Cisco Configuration Engine server.
- Step 5** Click **Add** to create an entry with the new router name.
The chosen name appears in the **Router Name** column in the lower part of the window.
- Step 6** In the **Local Networks** column, click the current entry in the field (this number indicates the number of local networks currently defined).

- Step 7** Specify the local Cisco Configuration Engine server network with the following values:
- In the **Network Name** field, enter the unique name as shown in “[Configuring the vrtd Daemon on the Prime Provisioning Master Machine](#)” section on page 2-10.
 - In the **Service** field, add the TIBCO port number for the Prime Provisioning installation (default: 7530).
 - The **Network Specification** field is optional. You can enter a description.
- Step 8** Click **Add Local Network**. The entered values appear in the corresponding columns in the lower section of the page.
- Step 9** Click on the entry just created. In this example, it is **prime**.
- Step 10** In the **Add Subject** field, enter **cisco.cns.>**.
- Step 11** Click **Add for Import and Export**. The entered values appear in the **Imported Subjects** and **Exported Subjects** columns in the lower part of the window.
- Step 12** If you are using Cisco Configuration Engine 1.3.2, 1.4, 1.5, 2.0, 3.0, or 3.5 in the **Subject** field in the lower part of the window, enter **cisco.mgmt.cns.>**, repeat [Step 11](#), and then proceed to [Step 13](#). If you are using Cisco Configuration Engine 1.3 or 1.3.1, then proceed to [Step 13](#).
- Step 13** Click the **routers** link in the left column.
- Step 14** In the **Local Networks** column, click the current entry in the field (this is at least **1** now, because you already added one local network).
- Step 15** Specify the local Cisco Configuration Engine network with the following values:
- In the **Local Network Name** field, add a unique name. For example: **ciscoconfigurationengine-eventBus**.
 - In the **Service** field, add the **CNS Event Bus Service Parameter** value defined in the setup of Cisco Configuration Engine server (default: 7500).
 - In the **Network Specification** field, leave it blank or enter the name of the Cisco Configuration Engine server.



Note If you encountered *any* error, check the check box for the row of information you want to remove, then click **Remove Marked Items**.

- Step 16** Click on the entry just created in the **Local Network Name** column.
- Step 17** In the **Add Subject** field in the upper part of the window, enter **cisco.cns.>**.
- Step 18** Click **Add for Import and Export**. The entered values appear in the **Imported Subjects** and **Exported Subjects** columns in the upper part of the window.
- Step 19** If you are using Cisco Configuration Engine 1.3.2, 1.4, 1.5, 2.0, 3.0, or 3.5 in the **Subject** field in the lower part of the window, enter **cisco.mgmt.cns.>**, repeat [Step 18](#), and then proceed to [Step 20](#). If you are using Cisco Configuration Engine 1.3 or 1.3.1, just proceed to [Step 20](#).
- Step 20** Click the **routers** link in the left column.
- Step 21** In the **Neighbors** column, click the current entry in the field (this number indicates the number of neighbors currently defined).
- Step 22** Add the following in the **Neighbors Configuration** window:
- In the **Neighbor Name** column, add the router name as automatically configured on the Prime Provisioning Master server, and verified in section “[Configuring the vrtd Daemon on the Prime Provisioning Master Machine](#).” This router name is **<isc_hostname>**.



Note It is very important that the **Neighbor Name** is the same as the **router** name configured on the Prime Provisioning Master server.

- b. In the **Hostname or IP addr** column, add the hostname or IP address of the Prime Provisioning Master server.
- c. In the **Remote** column, add the **Port** number for the **Local Endpoint** defined on the Prime Provisioning Master server as shown in the section “[Configuring the rvrD Daemon on the Prime Provisioning Master Machine.](#)”
- d. In the **Local** column, add the **Port** number for **Remote Endpoint** defined on the Prime Provisioning Master server, as shown in section “[Configuring the rvrD Daemon on the Prime Provisioning Master Machine.](#)”

Step 23 Click **Add Active [all]**.

A good indication that the connection is established is when the new name in the **Neighbor Name** column appears as a hyperlink in the bottom of the window. It takes a few seconds for this to occur. Also, it is recommended to click **Refresh** a few times to see the hyperlink.



Note If you encountered *any* error, check the check box for the row of information you want to remove, then click **Remove Marked Items**.

Checking Router Configurations Overview

The Cisco IOS image is needed for the routers used with the Cisco Configuration Engine functionality (that is, the CNS transport mechanism and/or the CNS Plug-and-Play feature). For Cisco Configuration Engine Release 1.3, the recommended Cisco IOS release is 12.2(8)T or later; for Cisco Configuration Engine Release 1.3.1, 1.3.2, 1.4, 1.5, 2.0, 3.0, or 3.5, the recommended Cisco IOS release is 12.2(11)T or later. Cisco IOS releases 12.3(1)T or later are supported only by Cisco Configuration Engine Releases 1.3.2, 1.4, 1.5, 2.0, 3.0, and 3.5.

Additionally, the router running a configuration must contain the following CNS commands:

1. **cns config partial** *<cisco configuration engine server IP address>* **80**

2. **cns event** *<cisco configuration engine server IP address>* **11011**

or

cns event *<cisco configuration engine server IP address>* **11011 keepalive** *<num. of seconds>*
<num. of trials>



Note The **keepalive** option makes sure the TCP connection between Cisco Configuration Engine and the router is alive at all times. It sends keepalive messages at *<num. of seconds>* intervals with *<num. of trials>* retries.

3. For IOS versions 12.3(1)T or later (12.0(27)S2 or later for Cisco 12000 (GSR) Series): **cns exec 80**

Also, the router startup configuration must contain the following two CNS commands:

1. **cns config initial** *<cisco configuration engine server IP address>* **event**

The **cns config initial** command should be configured in the startup configuration of the Cisco IOS device or router. It triggers the router to pick up and apply any initial configuration that might be waiting for it on the Cisco Configuration Engine server. After the **cns config initial** command is executed, this command is automatically removed. The recommendation is to include the **cns config partial** command in the initial configuration that is waiting on Cisco Configuration Engine. If a **no persist** option is used, the router does not perform a **write-mem**, thus keeping the startup configuration from being overwritten.

2. **cns event** *<cisco configuration engine server IP address>* **11011**

or

cns event *<cisco configuration engine server IP address>* **11011** **keepalive** *<num. of seconds>*
<num. of trials>



Note The **keepalive** option makes sure the TCP connection between Cisco Configuration Engine and the router is alive at all times. It sends keepalive messages at *<num. of seconds>* intervals with *<num. of trials>* retries.

Different IOS versions can support additional CNS commands or different formats of the same CNS command. See the Cisco Configuration Engine software documentation for more details on the other possible CNS commands and their options.