C H A P T E R **6**

WatchDog Commands

The WatchDog is responsible for bootstrapping Prime Provisioning and starting the necessary set of server processes. In addition, the WatchDog monitors the health and performance of each server to ensure it is functioning properly. In the event of a software error that causes a server to fail, the WatchDog automatically restarts the errant server.

The WatchDog is a background daemon process that is automatically installed as part of the installation procedure for Prime Provisioning. After the installation procedure has completed, WatchDog is started automatically. You can execute the **startwd** command to run the WatchDog after the installation. The WatchDog can be configured to automatically start any time the machine is rebooted.

In addition to the commands that are specified in this chapter, in the product you can choose **Administration > Control Center > Hosts** and from there you can start, stop, restart, and view log files for the individual Prime Provisioning servers.

This chapter provides the description, syntax, and arguments (listed alphabetically) for the following WatchDog commands:

startdb Command

This section provides the description and syntax for the **startdb** command.

Description

The **startdb** command starts the database.

Syntax

Go to **PRIMEP_HOME** and execute the following command:

./prime.sh startdb

The **startdb** command has no arguments and starts the database.

The location of **startdb** is: <Prime Provisioning *Directory*>/**bin**.

> **Note**   Do *not* run **startdb** in the background. Do *not* enter **startdb &**.

startns Command

This section provides the description and syntax for the **startns** command.

Description

The **startns** command starts the name server. The **orbd** process provides the name server functionality. **orbd** (from JDK) is required, but **startwd** starts it if it is not already running. The **startns** and **stopns** commands deal with **orbd**.

Syntax

Go to **PRIMEP_HOME** and execute the following command:

./prime.sh startns

The **startns** command has no arguments and starts the name server.

The location of **startns** is: <Prime Provisioning *Directory*>/**bin**.

startwd Command

This section provides the description and syntax for the **startwd** command.

Description

The **startwd** command starts the WatchDog and all Prime Provisioning processes. The **startwd** command includes the functionality of **startdb** (see the "startdb Command" section on page 6-1) and **startns** (see the "startns Command" section on page 6-2). Executing this command is a necessary procedure and occurs automatically as part of the installation. Use this **startwd** command after issuing a **stopwd** command to restart the WatchDog.

If for some reason the Prime Provisioning host is stopped, either inadvertently or by issuing the **stopwd** command, it can be restarted by using the **startwd** command.

Syntax

Go to **PRIMEP_HOME** and execute the following command:

./prime.sh startwd

The **startwd** command has no arguments and starts the WatchDog only for the machine where it is executed.

The location of **startwd** is: <Prime Provisioning *Directory*>**/bin**

Note   Do *not* run **startwd** in the background. Do *not* enter **startwd &**.

stopall Command

This section provides the description and syntax for the **stopall** command.

Description

The **stopall** command stops the database, name server, and WatchDog on the machine on which it is run. The **stopall** command includes the functionality of **stopdb -y** (see the "stopdb Command" section on page 6-3), **stopns -y** (see the "stopns Command" section on page 6-4), and **stopwd -y** (see the "stopwd Command" section on page 6-4). Normally this is only necessary before installing a new version of Prime Provisioning.

Syntax

Go to **PRIMEP_HOME** and execute the following command:

./prime.sh stopall

Caution   There is no **-y** parameter. Therefore, everything stops without the ability to cancel.

The location of **stopall** is: <Prime Provisioning *Directory*>**/bin**.

stopdb Command

This section provides the description and syntax for the **stopdb** command.

Description

The **stopdb** command stops the database.

Syntax

Go to **PRIMEP_HOME** and execute the following command:

./prime.sh stopdb [**-y**]

where:

-y indicates not to prompt before shutdown. If **-y** is not specified, you are prompted with the following message: "Are you absolutely sure you want to stop the database?" You are then prompted to reply **yes** or **no**.

The location of **stopdb** is: <Prime Provisioning *Directory*>/**bin**.

stopns Command

This section provides the description and syntax for the **stopns** command.

Description

The **stopns** command stops the name server. The **startns** and **stopns** commands deal with **orbd**.

Syntax

Go to **PRIMEP_HOME** and execute the following command:

./prime.sh stopns [**-y**]

where:

-y indicates not to prompt before shutdown. If **-y** is not specified, you are prompted with the following message: "Are you absolutely sure you want to stop the nameserver?" You are then prompted to reply **yes** or **no**.

The location of **stopns** is: <Prime Provisioning *Directory*>/**bin.**

stopwd Command

This section provides the description and syntax for the **stopwd** command.

Description

The **stopwd** command stops the WatchDog and all Prime Provisioning processes other than the name server and the database.

Syntax

Go to **PRIMEP_HOME** and execute the following command:

./**prime.sh stopwd** [**-y**]

where:

-y indicates not to prompt before shutdown. If **-y** is not specified, you are prompted with the following message: "Are you absolutely sure you want to stop the watchdog and all of its servers? Other users may be using this system as well. No activity (for example: collections, performance monitoring, provisioning) occurs until the system is restarted." You are then prompted to reply **yes** or **no**.

The location of **stopwd** is: <Prime Provisioning *Directory*>/**bin**.

wdclient Command

> **Note** Before running the wdclient command in Prime Provisioning, you need to source the environment variable with the following command: **"source $PRIMEF_HOME/bin/vpnenv.sh"**.

This section provides the description, syntax, and options (listed alphabetically) for the **wdclient** subcommands. These subcommands are diagnostic tools. This section also describes the column format of the output of each of the subcommands.

> **Note** The location of **wdclient** is: <Prime Provisioning *Directory*>/**bin**.

The following are the **wdclient** subcommands:

> **Note** If you enter **wdclient -help**, you receive a listing of all the wdclient subcommands.

wdclient disk Subcommand

This section provides the description and syntax for the **wdclient disk** subcommand.

Description

The **wdclient disk** subcommand gives the disk space statistics for the directories where Prime Provisioning is installed.

Syntax

wdclient disk

wdclient group *<group_name>* Subcommand

This section provides the description and syntax for the **wdclient group *<group_name>*** subcommand.

Description

The **wdclient group *<group_name>*** subcommand lists the servers in the specified server group. Server groups provide a convenient way to start or stop a group of servers with a single command.

Syntax

wdclient group *<group_name>*

where:

<group_name> is the name of a server group chosen from the list displayed by the **wdclient groups** command.

wdclient groups Subcommand

This section provides the description and syntax for the **wdclient groups** subcommand.

Description

The **wdclient groups** subcommand lists all the active server groups.

Syntax

wdclient groups

wdclient health Subcommand

This section provides the description and syntax for the **wdclient health** subcommand.

Description

The **wdclient health** subcommand indicates whether all the servers are stable.

Syntax

wdclient health

wdclient restart Subcommand

This section provides the description and syntax for the **wdclient restart** subcommand.

Description

The **wdclient restart** subcommand restarts one or more servers. Any dependent servers are also restarted.

> **Note**   It is not necessary to restart servers in a properly functioning system. The **wdclient restart** command should only be run under the direction of Cisco Support.

Syntax

wdclient restart [**all** | *<server_name>* | **group** *<group_name>*]

where you can choose one of the following arguments:

all is all servers. This is the default if no argument is specified.

<server_name> is the name of a server chosen from the list displayed by the **wdclient status** command. See Table 6-1, "Servers and Their Functions," for server descriptions.

group *<group_name>* where, *<group_name>* is the name of a server group chosen from the list displayed by the **wdclient groups** command.

wdclient start Subcommand

This section provides the description and syntax for the **wdclient start** subcommand.

Description

The **wdclient start** subcommand starts one or more servers. Other servers that depend on the specified server(s) might also start.

> **Note**   It is not necessary to stop and start servers in a properly functioning system. The **wdclient start** command should only be run under the direction of Cisco Support.

Syntax

wdclient start [**all** | *<server_name>* | **group** *<group_name>*]

where you can choose one of the following arguments:

all is all servers. This is the default if no argument is specified.

<server_name> is the name of a server chosen from the list displayed by the **wdclient status** command. See Table 6-1, "Servers and Their Functions," for server descriptions.

group *<group_name>* where, *<group_name>* is the name of a server group chosen from the list displayed by the **wdclient groups** command.

wdclient status Subcommand

This section provides the description, syntax, and information produced for the **wdclient status** subcommand.

Description

The **wdclient status** subcommand lists all the servers and their states. See Table 6-1 on page 6-8, "Servers and Their Functions," for server descriptions. See Table 6-2 on page 6-9, "Valid States," for the list of all the states.

Syntax

wdclient [**-poll** *<seconds>*] **status**

where:

-poll *<seconds>* is an optional parameter. *<seconds>* is the number of seconds. A number other than zero indicates that when new status data is available it is displayed every *<seconds>* seconds, where *<seconds>* is the specified number of seconds. The default **-poll** value is zero (0), which shows the status just once.

Information Produced: Name Column

The **Name** column provides the name of each of the servers. Table 6-1 provides a list of the servers and a description of the function that each server provides.

Table 6-1   Servers and Their Functions

Server	Function
dbpoller	Monitors database server.
discovery	Devices and Service Discovery Engine.
httpd	Web server.
nspoller	Monitors name service.
rgserver	Executes various Prime Provisioning traffic engineering computations, such as tunnel repairing.

> **Note**   The processes that no longer exist includes dispatcher, lockmanager, scheduler, and worker.

Information Produced: State Column

The **State** column provides the current state of the server. Table 6-2 provides a description of each of the states in normal progression order.

Table 6-2   Valid States

State	Description
start_depends	This server has been asked to start, but is waiting for servers it depends on to start. After all dependent servers have started, this server transitions to the state of starting.
starting	This server is currently starting. After a successful heartbeat occurs, this server transitions to the state of started.
started	This server is currently started and running.
stop_depends	This server is supposed to be stopped, but it is waiting for servers it depends on to be stopped first.
stopping_gently	This server is in the process of stopping in a gentle fashion. That is, it was notified that it is to stop.
stopping_hard	This server is in the process of being killed because either it did not have a way to stop gently or because the gentle stop took too long.
stopped	This server is stopped. The WatchDog either starts it again or disables it if it has been frequently dying.
disabled_dependent	This server is disabled because one or more servers it depends on are disabled. If all servers it depends on are started, this server automatically starts.
disabled	This server is disabled and must be manually restarted.
restart_delay	This server is delaying before restarting. There is a short delay after a server stops and before it is restarted again.

Information Produced: Gen Column

The **Gen** column provides the generation of the server. Each time the server is started, the generation is incremented by 1.

Information Produced: Exec Time Column

The **Exec Time** column provides the date and time the server was last started.

Information Produced: Success Column

The **Success** column provides the number of successful heartbeats since the server was last started. Heartbeats are used to verify that servers are functioning correctly.

Information Produced: Missed Column

The **Missed** column provides the number of missed heartbeats since the server was last started.

A few missed heartbeats could simply indicate the system was busy. However, more than a couple of missed heartbeats per day could indicate a problem. See the logs to diagnose the reason.

Three missed heartbeats in a row is the default for restarting the server.

wdclient stop Subcommand

This section provides the description and syntax for the **wdclient stop** subcommand.

Description

The **wdclient stop** subcommand stops one or more servers. Other servers that depend on the specified servers also stop.

Note   It is not necessary to stop servers in a properly functioning system. The **wdclient stop** command should *only* be run under the direction of Cisco Support.

Syntax

wdclient stop [**all** | *<server_name>* | **group** *<group_name>*]

where you can choose one of the following arguments.

all is all servers. This is the default if no argument is specified.

<server_name> is the name of a server chosen from the list displayed by the **wdclient status** command. See Table 6-1, "Servers and Their Functions," for server descriptions.

group *<group_name>* is the name of a server group chosen from the list displayed by the **wdclient groups** command.