



# Administering DFM

---

These topics describe tasks you may need to perform when administering DFM:

- [Security Issues, page 11-1](#)
- [How the DFM Broker Works, page 11-8](#)
- [Domain Manager Settings, page 11-10](#)
- [DFM and CiscoWorks Processes, page 11-12](#)

## Security Issues

DFM is designed to maintain the security of your system if it is installed and used as described in *Installation and Setup Guide for Device Fault Manager*. Be aware of the guidelines in these topics:

- [File Ownership and Protection, page 11-2](#)
- [Changing Usernames and Passwords for DFM and DFM Clients, page 11-2](#)
- [User Access, page 11-5](#)
- [Ports and Protocols Used by DFM, page 11-6](#)
- [Output Files, page 11-7](#)
- [DFM and Browsers, page 11-8](#)

## File Ownership and Protection

In DFM 1.0, DFM set file ownership to user bin. All DFM files are now installed with owner CASUSER. ROOT or privileged users can create, delete, or modify the files installed in *NMSROOT*. *NMSROOT* is the directory where CiscoWorks is installed (for Solaris, the default value for *NMSROOT* is /opt/CSCOpX, and for Windows, the default value is C:\Program Files\CSCOpX).

**Caution**

---

Do not change the protection of any file or directory to be less restrictive. You may, if you wish, make the protections more restrictive.

---

**Note**

---

On Windows, file protections are not enforced on FAT partitions.

---

## Changing Usernames and Passwords for DFM and DFM Clients

DFM provides extended security when DFM clients (such as adapters and consoles) connect to the server by validating the client connections with a username and password. The username and password are set when DFM and DFM clients are installed and are stored in *NMSROOT/conf/serverConnect.conf* and *clientConnect.conf*. Both files are stored on the server machines; only the *clientConnect.conf* file is stored on DFM remote client machines.

You can change the username and password by using the *DFMConnect.pl* command (available from *NMSROOT/cgi-bin/DFMConnect.pl*). The username and password for the DFM server and its clients must be identical, and you must restart the server and its clients after making any changes.

[Table 11-1](#) describes the options and arguments for the *DFMConnect.pl* command.

**Table 11-1 Options and Arguments to the DFMConnect.pl Command**

Option	Description
-s -show	Display the current username and password settings in the serverConnect.conf and clientConnect.conf files (run this command on the DFM server).
-c -show	Display current username and password settings in the clientConnect.conf file (run this command on the DFM client).
-s -r <i>username password</i>	Replace the current username and password settings in the serverConnect.conf and clientConnect.conf files (on the DFM server) with <i>username</i> and <i>password</i> . Valid usernames and passwords must contain 4-20 characters. Valid characters are: [a-z], [A-Z], [0-9], -, _, \$
-c -r <i>username password</i>	Replace the current username and password settings in the clientConnect.conf file (on the DFM client) with <i>username</i> and <i>password</i> . The username and password must be identical to those on the DFM server. Valid usernames and passwords must contain 4-20 characters. Valid characters are: [a-z], [A-Z], [0-9], -, _, \$

To change the username and passwords on a server and client machine, use this procedure:

**Step 1** Log in to the DFM server.

**Step 2** From the DFM server, change the username and password for the serverConnect.conf and clientConnect.conf files by entering:

```
DFMConnect.pl -s -r username password
```

For example, to change the username to cisco and the password to sanjose:

```
DFMConnect.pl -s -r cisco sanjose
```

- Step 3** From the DFM server, restart the necessary processes by doing one of the following:
- Restart the CiscoWorks server (this method is useful if you do not have many CiscoWorks applications running):
    - On Solaris:

```
# /etc/init.d/dmgtld stop
# /etc/init.d/dmgtld start
```
    - On Windows:

```
C:\> net stop crcmdmgtld
C:\> net start crcmdmgtld
```
  - Restart the following processes by selecting **Server Configuration > Administration > Process Management** (this method if you have many CiscoWorks applications running). Be sure to restart the processes in the following order:
    1. JRunProxyServer
    2. DFMBroker
    3. DFMServer
    4. DFMChangeProbe (if you are using the RME Adapter)
    5. Run ovstop and ovstart (if you are using the HPOV-NetView Adapter)
- Step 4** If you are using any remote clients (adapters), log into the remote DFM client machine.
- Step 5** From the DFM client, change the username and password for the clientConnect.conf file by entering (use the same username and password as in Step 2):
- ```
DFMConnect.pl -c -r username password
```

- Step 6** From the DFM client, restart the necessary processes by doing one of the following:
- If CiscoWorks is running on the DFM client, restart the CiscoWorks server (you may want to use this method if you do not have many CiscoWorks applications running):
    - On Solaris:

```
# /etc/init.d/dmgtld stop
# /etc/init.d/dmgtld stop
```
    - On Windows:

```
C:\> net stop crcmdmgtld
C:\> net start crcmdmgtld
```
  - If you have many CiscoWorks applications running, restart the necessary individual processes:
    - DfmChangeProbe (if you are using the RME Adapter).
    - Run ovstop and ovstart (if you are using the HPOV-NetView Adapter)
- Step 7** Repeat Steps 4-6 for any other DFM clients that are using remote adapters.
- 

## User Access

DFM servers grant access to clients who can form a TCP connection to their port. Clients include the Monitoring and Administration Consoles, as well as adapters. You can control this access in the following ways:

- [Controlling User Access with the --accept Option, page 11-6](#)
- [Controlling User Access through Port Control, page 11-6](#)



### Note

To view the current access assigned to users, run a CiscoWorks Permissions Report by selecting **Server Configuration > Setup > Security > Permissions Report**.

---

## Controlling User Access with the --accept Option

The --accept option is provided by the DFM servers and broker. Use this option to control the hosts that can connect to a DFM server. Host connections include DFM consoles, adapters, and utilities. Processes running on the same host as a server do not have special access—the server's host must be listed to allow the processes to access the host.

**Note**

---

If you specify registration options using pdcmd, you must re-run your command whenever the daemon manager restarts.

---

If an unlisted host attempts to initiate a connection, the server rejects the attempt, writes a message to the server's stderr file, and sends the message to the system logging facility.

For information on using this option, refer to the [“Registering and Unregistering the DFM Processes Using pdcmd”](#) section on page 11-15.

## Controlling User Access through Port Control

You can request DFM servers to listen on a specified port (rather than a default port.) Contact the Technical Assistance Center if you wish to set up such a configuration. Note that DFM does not support firewalls between clients and servers. The ports that DFM uses are listed in the [“Ports and Protocols Used by DFM”](#) section on page 11-6.

## Ports and Protocols Used by DFM

DFM uses the following ports and protocols.

- Ports:
  - 161 (SNMP) - UDP port
  - 162 (SNMP trap) - UDP port

- 9000 (SNMP trap, if port 162 is occupied/DfmServer) - UDP port

**Note**

---

You can configure DFM to always use a privileged port after rebooting. This is useful when you have removed HP OpenView and NetView, and you are sure that no other NMS will need port 162. To do this, register the DfmServer process using the --privopen option, and described in the [“Registering and Unregistering the DFM Processes Using pdcmd”](#) section on page 11-15.

---

- 9002 (DynamID authentication/DfmBroker) - TCP port
- Protocols:
  - SNMP
  - ICMP
  - TCP/IP
  - SMTP

You can configure DFM to always use a privileged port after rebooting. This is useful when removing HP OpenView and NetView, and you are sure that no other NMS will need port 162. To do this, register the DfmServer process using the --privopen option, and described in the [“Registering and Unregistering the DFM Processes Using pdcmd”](#) section on page 11-15.

## Output Files

DFM is designed so that its output files can be written only into the DFM installation tree, which consists of the directories under *NMSROOT*/objects/smarts. (*NMSROOT* is the directory where CiscoWorks is installed (for Solaris, the default value for *NMSROOT* is /opt/CSCOpX, and for Windows, the default value is C:\Program Files\CSCOpX).)

## DFM and Browsers

You can use DFM consoles within a Web browser. If you do this, do not use the same running browser to both access untrusted Web sites and manage your network.

If you continue fault monitoring after exiting CiscoWorks, you should close your web browser so other users cannot access your session.

## How the DFM Broker Works

A client application, such as a console or an adapter, utilizes the broker to determine where the domain manager is running. When the domain manager starts, it registers the hostname of the machine it is running on and the TCP port it is listening on with the broker. Clients retrieve this information from the broker so that they can communicate with the domain manager.

## Information the DFM Broker Maintains

The broker maintains the following information:

- The name of the domain manager and the host and TCP port it is running on.
- The status of each domain manager.
  - Running indicates that the broker is able to communicate with the domain manager.
  - Dead indicates that the domain manager exited unexpectedly or is unreachable. When the domain manager properly shuts down, it notifies the broker and the broker removes it from the list.
  - Unknown indicates that the broker was restarted and that it is querying the domain manager to determine its state.

The broker checks the status of the domain manager every five minutes by connecting to it and determining whether a DFM process is running correctly. If the broker is unable to connect or the process is not running, it changes the status of the domain manager to dead.

- The process ID of the domain manager. This is the process ID assigned by the host's operating system. In some cases when the broker is restarted, the process ID of the domain manager is set to zero to indicate the broker does not know the process ID of the domain manager.
- The last time the state of the domain manager changed. This value is set when the domain manager registers with the broker and is updated if the broker determines that the domain manager is dead. When the broker restarts, it changes the status of the domain manager (marked Unknown to Running or Dead).

## How DFM Clients Find the Broker

Clients follow these steps to determine the broker's location.

1. The client checks to see if the broker's location was passed as a command line option during startup.
2. The client checks the value of the SM\_BROKER environment variable. The value of the SM\_BROKER variable is normally set during installation.
3. The client checks if the broker is running on the host *broker\_machine* and listening on TCP port 9002.



---

**Note**

The hostname *broker\_machine* is usually an alias, such as a DNS CNAME.

---

4. If the alias *broker\_machine* is not defined, the client tries to use localhost:9002.

# Domain Manager Settings

This section describes the parameters related to the operation of a domain manager. Changing these parameters can substantially affect analysis results. For normal operation, the default settings should be sufficient.

**Caution**

---

Any change you make to the domain manager affect all consoles connected to the domain manager.

---

The two commands described in the [“Special Commands”](#) section on page 11-10 are listed under the Domain menu and in the domain manager’s pop-up menu. During normal operation, you will not need to use these commands.

## Special Commands

Domain manager special commands are:

- **Recompute Codebook**

Recompute Codebook tells the domain manager to recompute the codebook because the managed system’s inventory has changed. Analysis rules (codebook) are automatically computed from the analysis model and the managed topology. DFM automatically recomputes the codebook when the topology changes. It is not necessary to perform this procedure during normal operation.

- **Correlate Now**

Correlate Now forces the domain manager to analyze events. This is useful when the analysis interval is several minutes long, you know something has happened, and you want the domain manager to analyze the results now.

## Domain Manager Properties

When the icon for the domain manager is selected, the Administration Console displays four tabs in the right panel: Correlation, Modules, Threads, and Inventory. The first three tabs control properties of the Domain Manager.

## Correlation Tab

Correlation parameters determine how DFM diagnoses problems.

- Correlation Interval determines how often the domain manager analyzes events to find fault conditions. When the interval is increased, a domain manager looks for fault conditions less often and, as a result, generates notifications less frequently. This value does not affect how often the domain manager generates symptom event notifications, only how often it generates compound notifications.
- Codebook Radius determines the level of noise (delayed, lost, or spurious events) a domain manager can tolerate and still guarantee correct analysis results. A higher number means that the server tolerates more noise (lost or spurious events). This also results in more notifications with a low certainty as the domain manager accounts for every possible cause.
- Correlation Radius determines how exact a match must occur between symptoms known to be caused by a fault condition and the events the domain manager monitors. A domain manager uses the model and network topology to compute the symptoms each fault condition is expected to cause. These are termed a *problem signature*. When the events that the domain manager monitors match a problem signature within a tolerance defined by the radius, it determines that the corresponding fault condition occurred.

When the radius is too high, the domain manager considers problems whose signatures are quite different from the monitored events, though it will give them a low certainty. When the radius is too small, the problem signatures and monitored events must match so closely that a single lost event can disrupt accurate analysis.

- Number of Problems determines how many concurrent fault conditions a domain manager should consider possible when it analyzes events. It is not unusual for more than one fault condition to occur in a given interval. When one fault condition does not match the codebook, the domain manager checks to see if two fault conditions caused the symptoms. Increasing the number of fault conditions increases the amount of time it takes a domain manager to analyze because it multiplies the possible combinations of fault conditions that must be considered.

The limitation on the number of concurrent fault conditions only applies to fault conditions with related symptoms. Sophisticated algorithms in the domain manager allow it to efficiently consider a large number of concurrent fault conditions whose symptoms are unrelated.

- Loss Symptom Probability is the likelihood that the domain manager will not receive notice of a symptom that actually occurred.
- Spurious Symptom Probability is the likelihood that the domain manager will falsely receive notice that a symptom has occurred when it actually has not.

## DFM and CiscoWorks Processes

These topics describe how to administer DFM-specific CiscoWorks processes:

- [DFM-Related CiscoWorks Processes, page 11-12](#)
- [Registering and Unregistering the DFM Processes Using pdcmd, page 11-15](#)

## DFM-Related CiscoWorks Processes

The following table provides a complete list of DFM-related CiscoWorks processes.

**Note**

---

You cannot stop or unregister the DfmServer process if any process that depends on it is running. You must first stop or unregister all dependent processes, and then stop or unregister the DfmServer process.

---

Table 11-2 DFM-Related CiscoWorks Processes

| Name            | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  | Dependency                              |
|-----------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------|
| DfmBroker       | <p>The DFM Broker maintains a registry about the DFM domain manager. The domain manager registers the following information with the broker when its initialization is complete:</p> <ul style="list-style-type: none"> <li>• Application name of the domain manager</li> <li>• Hostname where the domain manager is running</li> <li>• TCP port at which the HTTP server is listening</li> </ul> <p>When a client needs to connect to the domain manager, it first connects to the broker to determine the hostname and TCP port where that server's HTTP service is listening. It then disconnects from the broker and establishes a connection to the domain manager.</p> | None                                    |
| DfmServer       | <p>DFM domain manager, a DFM program that provides backend services for DFM. Services include SNMP data retrieval and event analysis (SNMP Trap Adapter). The DfmServer log is <i>NMSROOT/objects/smarts/logs/DFM.log</i>.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                               | DfmBroker                               |
| DfmFileNotifier | <p>Logs DFM analysis results into ASCII files (File Notifier Adapter).</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   | DfmServer                               |
| DfmTrapNotifier | <p>Converts DFM notifications into SNMP traps (Trap Notifier Adapter).</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   | DfmServer                               |
| DfmMailNotifier | <p>Monitors user-specified alarms and events and automatically emails notifications to specified recipients (Mail Notifier Adapter).</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     | DfmServer                               |
| DfmChangeProbe  | <p>Monitors messages from Essentials processes and forwards inventory changes to the DFM domain manager (RME Adapter).</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   | EssentialsDbEngine, EssentialsDbMonitor |

Table 11-2 DFM-Related CiscoWorks Processes (continued)

| Name           | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                               | Dependency |
|----------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------|
| DfmisdnUpgrade | <p><b>Note</b> This process is provided in DFM 1.2 with IDU 1.2.5 (or later).</p> <p>At IDU installation, checks whether the DFM inventory must be modified to support new ISDN groups (introduced in DFM 1.2 IDU 1.2.5) and updates the inventory, if required. (Except during installation, this process should be in the state “Administrator has shut down this server.”)</p> <p>The DfmisdnUpgrade log is <i>NMSROOT/objects/smarts/logs/DFMisdnUpgrade.log</i>.</p> | DfmServer  |

To stop (or start) a CiscoWorks process:

- 
- Step 1** Log on to CiscoWorks as an administrator.
- Step 2** Select **Server Configuration > Administration > Process Management > Stop Process**. The Stop Process window opens.




---

**Note** If a process is not listed, it has not yet been started.

---

- Step 3** In the Stop Process window, locate the process you want to stop in the Process drop-down list.




---

**Note** The DFM installation procedure sets DfmServer to start automatically, so it is normally listed. When you stop the DfmServer process, any users attached to DFM will be detached. Use the Attach button to re-attach after the DfmServer process is restarted.

---

- Step 4** Select the process you want to stop and click the **Finish** button.
- Step 5** To restart the process, select **Server Configuration > Administration > Process Management > Start Process**. The Start Process window opens.

- Step 6** In the Start Process window, locate the process you want to start in the Process drop-down list.
- Step 7** Select the process you want to start and click the **Finish** button.
- 

## Registering and Unregistering the DFM Processes Using pdcmd

You can use pdcmd to manually unregister and reregister DFM processes with the CiscoWorks daemon manager. This is useful when you want to do any of the following:

- Specify clients that can connect to DFM.
- Configure adapters to restart automatically whenever the DFM server stops and restarts.
- Configure the DFM server to use a privileged port.

Because these commands are complex, be sure to refer to the examples in these sections:

- [Example: Specifying Clients That Can Connect to DFM, page 11-19](#)
- [Example: Configuring the DFM Server to Use a Privileged Port, page 11-20](#)

Before registering a process, you must unregister the related processes in this order:

1. Unregister all processes that depend on the DfmServer process (such as notification adapter processes).
2. Unregister the DfmServer process.
3. Unregister the DfmBroker process.

Use the following syntax when unregistering DFM processes (for Windows, the command is pdcmd.exe):

```
NMSROOT/bin/pdcmd -u process
```

When you reregister the process, specify all options in the same command instance. If you enter the `pdcmd` multiple times, only the last instance is used. Register the processes in the following order:

1. Register the `DfmBroker` process.
2. Register the `DfmServer` process.
3. Register all processes that depend on `DfmServer` (such as notification adapter processes).

Use the following syntax to reregister the DFM processes. (Refer to [Table 11-3](#) for information about the options and arguments).

```
NMSROOT/bin/pdcmd -r DfmBroker -e path -f arguments
NMSROOT/bin/pdcmd -r DfmServer -e path -d depends -f arguments
NMSROOT/bin/pdcmd -r adapter_process -d DfmServer
```

**Note**


---

To view the default settings for a process, enter `NMSROOT/bin/pdreg -l process`.

---

**Note**


---

If you specify registration options using `pdcmd`, you must re-run your command whenever the daemon manager restarts.

---

**Table 11-3 Options to *pdcmd***

| Option            | Description and Arguments                                                                                                                                                                                                                                     |
|-------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| -u <i>process</i> | Unregister <i>process</i> . The processes are listed in <a href="#">Table 11-2 on page 11-13</a> .                                                                                                                                                            |
| -r <i>process</i> | Register <i>process</i> to CiscoWorks daemon manager and start <i>process</i> whenever the dependent (parent) process starts (as described in the -d <i>depends</i> option). The processes are listed in <a href="#">Table 11-2 on page 11-13</a> .           |
| -e <i>path</i>    | Process binary path. <i>path</i> should be:                                                                                                                                                                                                                   |
|                   | <ul style="list-style-type: none"> <li>• DFM broker: <i>MSROOT/objects/smarts/bin/brstart</i></li> <li>• DFM server: <i>NMSROOT/objects/smarts/bin/sm_server</i></li> <li>• DFM notification adapters: <i>NMSROOT/objects/smarts/bin/sm_notify</i></li> </ul> |
| -d <i>depends</i> | Process dependency. <i>depends</i> should be:                                                                                                                                                                                                                 |
|                   | <ul style="list-style-type: none"> <li>• DFM server: <i>DfmBroker</i></li> <li>• DFM notification adapters: <i>DfmServer</i></li> </ul>                                                                                                                       |

Table 11-3 Options to `pdcmd` (continued)

| Option                                        | Description and Arguments                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|-----------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| -f " <i>arguments</i> "                       | DFM-specific arguments, enclosed in one set of quotes. <i>arguments</i> can be the following:                                                                                                                                                                                                                                                                                                                                                                                                                     |
| --accept <i>host1,host2...</i>                | (Optional.) Comma-separated list of hostnames or IP addresses specifying clients which can connect to the server. (The DFM server does not use reverse lookups to determine names of connecting host. If you specify clients as hostnames, be sure the hostname is in DNS, especially if you are using DHCP. If you want to specify localhost, use the host name or IP address, not <i>localhost</i> ; refer to the <a href="#">“Example: Specifying Clients That Can Connect to DFM”</a> section on page 11-19.) |
| --privopen= <i>open-list</i>                  | (Optional.) Specify the privileged ports and protocol which DfmBroker or DfmServer may open. <i>open-list</i> can be comma-separated list of the following (IP:protocol is always required):<br><br>TCP: <i>port</i> , UDP: <i>port</i> , IP: <i>protocol</i><br><br>The defaults for <i>open-list</i> depend on whether DFM is using a reserved port:                                                                                                                                                            |
| --privopen=IP:1                               | Default if reserved port is not being used.                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| --privopen=IP:1,<br>UDP: <i>reserved_port</i> | Default if reserved port is being used (normally 162).                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| --ouput= <i>file</i>                          | (Required.) Name of process output file. <i>file</i> should be: <ul style="list-style-type: none"> <li>• DfmServer: DFM</li> <li>• DfmFileNotifier: sm_file_notifier</li> <li>• DfmMailNotifier: sm_mail_notifier</li> <li>• DfmTrapNotifier: sm_trap_notifier</li> </ul>                                                                                                                                                                                                                                         |
| --port= <i>port</i>                           | (DfmBroker only.) DFM broker port. <i>port</i> should always be 9002.                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| --restore= <i>file</i>                        | (DfmBroker only.) Restore broker state from backup file. <i>file</i> should always be:<br><br>--restore=NMSROOT/objects/smarts/conf/broker.rps                                                                                                                                                                                                                                                                                                                                                                    |

## Example: Specifying Clients That Can Connect to DFM

This example shows how to configure DFM to only accept client connections from the hostnames `lucy` and `ethel`. In this case you must unregister and reregister the DFM broker, server, and notification adapter processes.

**Note**

To allow connections from processes running on the same host, specify the host's name—do not use “localhost.” This is because connections made using the DFM Broker will appear to come from the DFM Broker's host. Only connections that explicitly specify “localhost” as the target address will appear to come from localhost. Such target addresses may result in configurations that forward incoming connections (such as through software that provides an encrypted tunnel).

**Step 1** Unregister the processes.**a.** Unregister the DFM notification adapters:

```
# NMSROOT/bin/pdcmnd -u DfmFileNotifier
# NMSROOT/bin/pdcmnd -u DfmMailNotifier
# NMSROOT/bin/pdcmnd -u DfmTrapNotifier
```

**b.** Unregister the DFM server process:

```
# NMSROOT/bin/pdcmnd -u DfmServer
```

**c.** Unregister the DFM broker process:

```
# NMSROOT/bin/pdcmnd -u DfmBroker
```

**Step 2** Re-register the processes, specifying the clients that can connect to the broker and server:**a.** For the DFM broker (the following command is one line):

```
# NMSROOT/bin/pdcmnd -r DfmBroker -e NMSROOT/objects/smarts/bin/brstart -f "--output
--port=9002 --accept=lucy,ethel --restore=NMSROOT/objects/smarts/conf/broker.rps"
```

- b. For the DFM server (the following command is one line):

```
# NMSROOT/bin/pdcmnd -r DfmServer -e NMSROOT/objects/smarts/bin/sm_server -d DfmBroker -f
"--bootstrap=DFM_bootstrap.conf --accept=lucy,ethel --output --name=DFM"
```




---

**Note** When specifying other options (such as `--privopen`) for `DfmServer`, use one `pdcmnd` instance. See the [“Example: Configuring the DFM Server to Use a Privileged Port”](#) section on page 11-20.

---

- c. For DFM notification adapters (the following commands are each one line):

```
# NMSROOT/bin/pdcmnd -r DfmFileNotifier -d DfmServer -e
NMSROOT/objects/smarts/bin/sm_notify -f "--adapter=filelog --output=sm_file_notifier"
```

```
# NMSROOT/bin/pdcmnd -r DfmMailNotifier -d DfmServer -e
NMSROOT/objects/smarts/bin/sm_notify -f "--adapter=mail --output=sm_mail_notifier"
```

```
# NMSROOT/bin/pdcmnd -r DfmTrapNotifier -d DfmServer -e
NMSROOT/objects/smarts/bin/sm_notify -f "--adapter=trap --output=sm_trap_notifier"
```

## Example: Configuring the DFM Server to Use a Privileged Port

This example shows how to configure DFM to use a privileged port.

- Step 1** Unregister any processes that depend on the `DfmServer` (such as the notification adapters).

```
# NMSROOT/bin/pdcmnd -u DfmFileNotifier
# NMSROOT/bin/pdcmnd -u DfmMailNotifier
# NMSROOT/bin/pdcmnd -u DfmTrapNotifier
```

- Step 2** Unregister the `DfmServer` process:

```
# NMSROOT/bin/pdcmnd -u DfmServer
```

- Step 3** Re-register the `DfmServer` process to use UDP port 162 and the IP protocol 1:

```
# NMSROOT/bin/pdcmnd -r DfmServer -e NMSROOT/objects/smarts/bin/sm_server -d DfmBroker -f
"--bootstrap=DFM_bootstrap.conf --privopen=IP:1,UDP:162 --output --name=DFM"
```

**Step 4** Reregister any processes that depend on DfmServer:

```
# NMSROOT/bin/pdcmnd -r DfmFileNotifier -d DfmServer -e
NMSROOT/objects/smarts/bin/sm_notify -f "--adapter=filelog --output=sm_file_notifier"

# NMSROOT/bin/pdcmnd -r DfmMailNotifier -d DfmServer -e
NMSROOT/objects/smarts/bin/sm_notify -f "--adapter=mail --output=sm_mail_notifier"

# NMSROOT/bin/pdcmnd -r DfmTrapNotifier -d DfmServer -e
NMSROOT/objects/smarts/bin/sm_notify -f "--adapter=trap --output=sm_trap_notifier"
```

---

If you also want DFM to accept only specific client connections, you must specify the `--accept` option when registering the DfmServer process (you do not have to do this for the adapter processes). The following example registers the DfmServer process to use UDP port 162 and IP protocol 1, *and* specifies that DFM can accept connections from hostnames `lucy` and `ethel`:

```
# NMSROOT/bin/pdcmnd -r DfmServer -e NMSROOT/objects/smarts/bin/sm_server -d DfmBroker -f
"--bootstrap=DFM_bootstrap.conf --accept=lucy,ethel --privopen=IP:1,UDP:162 --output
--name=DFM"
```

