HP Operations Manager

Software Version: 9.22
Linux and UNIX operating systems

Administrator's Reference Guide

Document Release Date: December 2016
Software Release Date: December 2016

Legal Notices

Warranty

The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

The information contained herein is subject to change without notice.

Restricted Rights Legend

Confidential computer software. Valid license from HP required for possession, use or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

Copyright Notice

Trademark Notices

Adobe™ is a trademark of Adobe Systems Incorporated.

Microsoft® and Windows® are U.S. registered trademarks of Microsoft Corporation.

UNIX® is a registered trademark of The Open Group.

This product includes an interface of the 'zlib' general purpose compression library, which is Copyright © 1995-2002 Jean-loup Gailly and Mark Adler.

Documentation Updates

The title page of this document contains the following identifying information:

- Software Version number, which indicates the software version.
- Document Release Date, which changes each time the document is updated.
- Software Release Date, which indicates the release date of this version of the software.

To check for recent updates or to verify that you are using the most recent edition of a document, go to: **https://softwaresupport.hpe.com**

This site requires that you register for an HP Passport and sign in. To register for an HP Passport ID, go to: **https://cf.passport.hpe.com/hppcf/createuser.do**

Or click the **the Register** link at the top of the HP Software Support page.

You will also receive updated or new editions if you subscribe to the appropriate product support service. Contact your HP sales representative for details.

Support

Visit the HP Software Support Online web site at: **https://softwaresupport.hpe.com**

This web site provides contact information and details about the products, services, and support that HP Software offers.

HP Software online support provides customer self-solve capabilities. It provides a fast and efficient way to access interactive technical support tools needed to manage your business. As a valued support customer, you can benefit by using the support web site to:

- Search for knowledge documents of interest
- Submit and track support cases and enhancement requests
- Download software patches
- Manage support contracts
- Look up HP support contacts
- Review information about available services
- Enter into discussions with other software customers
- Research and register for software training

Most of the support areas require that you register as an HP Passport user and sign in. Many also require a support contract. To register for an HP Passport ID, go to:

https://cf.passport.hpe.com/hppcf/createuser.do

To find more information about access levels, go to:

https://softwaresupport.hpe.com/web/softwaresupport/access-levels

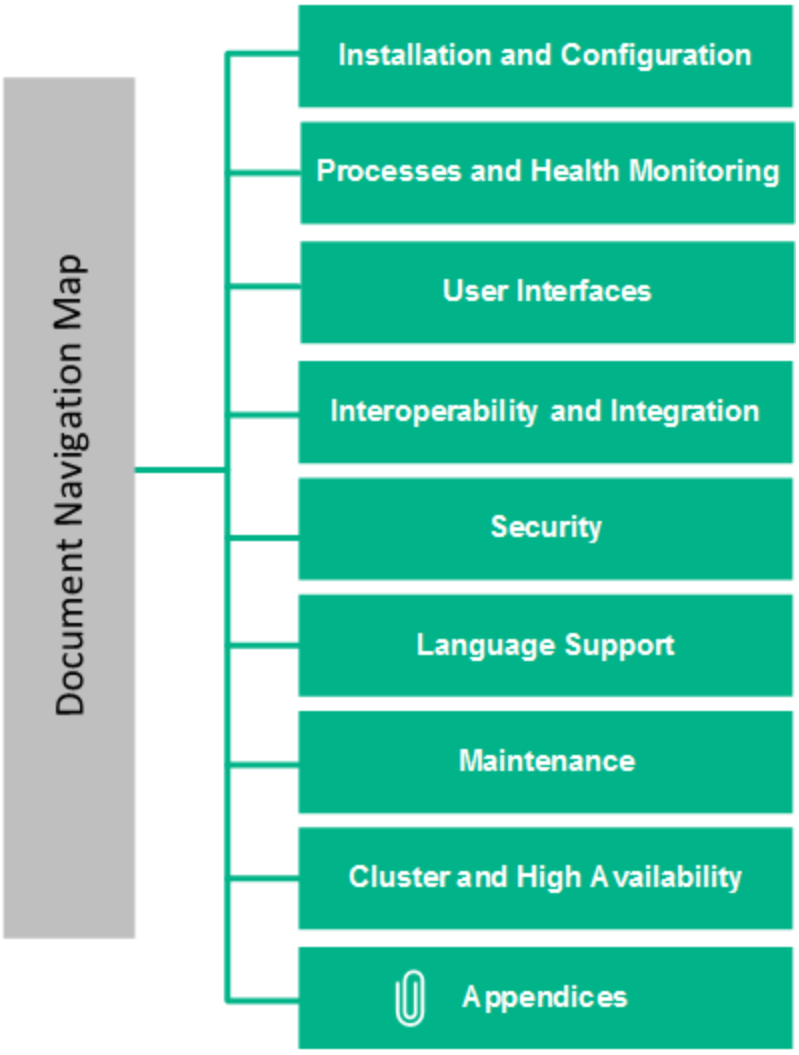
HP Software Solutions Now accesses the HPSW Solution and Integration Portal Web site. This site enables you to explore HP Product Solutions to meet your business needs, includes a full list of Integrations between HP Products, as well as a listing of ITIL Processes. The URL for this Web site is **https://softwaresupport.hpe.com**

Contents

About This Document

The Administrator's Reference Guide provides information required to administer and maintain the HP Operations Manager (HPOM) managed environment. The content of this document includes HPOM configuration and processes, overview of the HPOM user interfaces, security measures, and HPOM maintenance in standalone and cluster environments.

Click each item in the navigation map below for more information on the categories.



Part I: Installation and Configuration

After you have successfully installed and configured the database and HP Operations Manager (HPOM) on the management server, you must configure the managed nodes and distribute HPOM configuration to the nodes.

You can also configure HPOM to communicate over a network using the IPv6 protocol.

For details about installation and configuration, see:

- "HP Operations Agent Installation on HPOM Managed Nodes" on page 34
- "HPOM Configuration" on page 56
- "Configuring HPOM Managed Nodes " on page 125
- "HPOM Software Distribution to Managed Nodes " on page 211
- "Configuring HPOM to Use the IPv6" on page 232

For information about installing and configuring database and HPOM, see the *HPOM Installation Guide for the Management Server*.

Chapter 1: HP Operations Agent Installation on HPOM Managed Nodes

In this Chapter

This chapter gives general instructions on how to install the HP Operations Manager (HPOM) agent software on the supported managed nodes. The information in this section covers the following topics:

The installation procedures assume that you have already installed and configured the database and HPOM on the management server, as described in the *HPOM Installation Guide for the Management Server*.

Installation Requirements

This section describes the operating system, hardware, and software requirements for installing HPOM agents on the managed nodes.

Operating System Requirements

For a detailed list of the specific versions of the various agent operating systems that are supported by HPOM, see the *HPOM Installation Guide for the Management Server*.

Hardware and Software Requirements

For details about the hardware and software requirements for each supported managed node platform, see the *HP Operations Agent Release Notes*.

Kernel Parameters

Before installing HPOM, make sure the kernel parameters are set correctly. Although system default values are normally sufficient, the log file encapsulator sometimes requires an increase in the setting for the maximum permitted number of open files.

For information about recommended kernel parameters, see the HP Operations agent documentation.

Communication Software

HPOM uses the HTTPS mechanism to communicate between the management server and the client nodes. HTTPS 1.1 based communication is the latest communication technology used for HP BTO Software products and allows applications to exchange data between heterogeneous systems. HTTP/SSL is the default communication type for new HPOM nodes.

Agent Installation Tips

This section describes tips for installing HPOM agents both on the management server and on managed nodes. There is also a section including tips for the installation of the HPOM agent on UNIX managed nodes in particular. The information in this section covers the following topics:

- "Agent Installation on Managed Nodes" on page 35
- "Agent Installation on UNIX Managed Nodes" on page 38

Agent Installation on Managed Nodes

When installing the HPOM agent on the managed nodes, note the following guidelines:

- Installation or removal:

  Avoid interrupting the agent-software installation or removal process on managed nodes. Interrupting either process causes a semaphore file to be left on the management server. As a result, you will not be able to restart the installation.

  If a semaphore file is created on the management server, remove the file manually by entering:

  ```
  # rm /var/opt/OV/share/tmp/OpC/mgmt_sv/inst.lock
  ```

  If you interrupt the agent-software installation (or removal) on the managed nodes at the time you are asked for a password, your terminal settings will be corrupted, and any commands that you type will not be echoed in the terminal.

  If your terminal settings are corrupted, you can reset the terminal by entering the following:

  ```
  # stty echo
  ```

Note: If the multiple `inst.sh` functionality is enabled, semaphore files are created at the following location:

`/var/opt/OV/share/tmp/OpC/mgmt_sv/inst.lock.<node_name>`

- Management-server software:

  If any managed node is still configured and has the HPOM agent software still in place, do not remove any of the management server software bundles from the management server, for example: `OVOPC-ORA` or `OVOPC`.

- Tape image:

  If another managed node of the type you are deinstalling is still configured and has the HPOM agent software installed, do not remove the managed node tape images (for example `OVOPC-CLT-ENG`) from the management server. If you remove the tape image, you will be unable to remove the HPOM agent software from the managed node.

- Installation target:

  Whenever possible, install the latest HPOM agent software version on all managed nodes. Installing the latest version enables the latest HPOM features to be used on those nodes.

- Node-name restrictions:

  You may not use the names `bin`, `conf`, `distrib`, `unknown`, and `mgmt_sv` for managed nodes. These names are used internally by HPOM, and therefore may not be used as names of other systems.

- Host aliases:

  Avoid using host aliases. Identical host aliases cause system problems.

- IP address:

  Identify managed nodes having more than one IP address. Specify the most appropriate address (for example, the IP address of a fast network connection) in the HPOM configuration. Verify that all other IP addresses of that managed node are also identified on the management server. Otherwise, messages from multiple IP address systems might not be forwarded by HPOM.

- Disk space:

  During installation on managed nodes, twice the amount of disk space normally required by HPOM is needed. This extra disk space is needed because the tape image is transferred to the managed node before it is uncompressed and unpacked.

  If you do not have enough disk space for HPOM in your UNIX file system, consider applying one or both of the following solutions:

  - Use a symbolic link.

    For example, for Solaris, enter the following:

```
# ln -s /mt1/OV /opt/OV
```

- Mount a dedicated volume.

- Long host names:

  Use long host names in your policies only when performing automatic actions or operator-initiated actions.

- Operating system versions:

  Do not upgrade or downgrade the operating system version of the management server or managed node to a version not supported by HPOM. For a list of supported operating system versions on the management server and on the managed nodes, see the product support matrix or the *HPOM Installation Guide for the Management Server*.

  You can also get a list of the installed agent packages and the operating systems the agents support at the time the package was published by running the following script on the management server:

  ```
  # /opt/OV/bin/OpC/agtinstall/opcversion
  ```

- System time:

  Verify that the system times of the management server and the managed nodes are synchronized. By synchronizing system times, you ensure that the time at which the message is generated on the managed node is earlier than the time at which the message is received on the management server.

- Passwords:

  Before you install the HPOM agent software for the first time, make sure you know the root passwords for all the managed nodes on which you want to install the agent software. Note that passwords are not required for updates to agent installations.

  On UNIX managed nodes, passwords are not required if a `.rhosts` entry exists for the root user or if the management server is included in `/etc/hosts.equiv` (HP-UX 11.x, Solaris).

- Network paths:

  There must be an existing route (network path) in both directions between the HPOM management server and the managed nodes.

- Software removal:

  If you want to move the HPOM management server from one host to another, or change the hostname (or IP address) of the HP Operations management server, you must first remove the HPOM agent software from all nodes managed by the HPOM server that you want to reconfigure. For more information about changing the name or IP address of the management server, see "Host Names and IP Addresses" on page 688 or the *HPOM Installation Guide for the Management Server*.

- Default operator functionality:

If you do not need the functionality of the HPOM default operator on your managed nodes (except for the management server), you can purge the related information. The purged information is recreated when you reinstall the HPOM-agent software.

On UNIX managed nodes:

- Erase the home directory of the user `opc_op`.
- Remove the `opc_op` entry from `/etc/passwd`.
- Remove the `opcgrp` entry from `/etc/group`.

- Program management with agent APIs:

  When you upgrade or reinstall HPOM-agent software on managed nodes, make sure that all programs and applications that use the `opcmsg(3)` or `opcmon(3)` API are stopped.

  Note: This statement applies to all HPOM agent APIs including the message-stream interface (MSI).

  These APIs as well as other APIs are stored in the HPOM shared library, which is overwritten during HPOM-agent software upgrade or reinstallation.

Agent Installation on UNIX Managed Nodes

When installing the HPOM agents on UNIX managed nodes, note the following general guidelines:

- Short system name:

  Make sure that `uname(1m)` (HP-UX) or `uname(1)`(Solaris) returns the short system name.
- Fully qualified system name:

  Configure the name service (`/etc/hosts` or DNS) so all name-service operations (for example, `nslookup`) are consistently resolved to the fully qualified system name. For example, `hostname` is not name-service related and may return the short hostname.
- Log directory:

  Removal of HPOM deletes any non-default log directory on UNIX systems. The following rules apply to the default log directory:

  - Managed nodes:

    Do not use the same logging directory for more than one managed node. Using the same log directory for multiple managed nodes can cause problems if the directory is NFS-mounted across several systems.
  - Other applications:

    Do not use the same log directory for both HPOM and other applications.

- Subdirectories:

  Do not create subdirectories other than the HPOM log directory for use by other applications or managed nodes.

- Security file:

  Make sure that the security file for `inetd` on the managed nodes allows connections with `remshd` or `ftpd` to the management server. For example, for HP-UX 11.x, use the following:

  `/var/adm/inetd.sec`

- Root user:

  If no `.rhosts` entry for `root` and no `/etc/hosts.equiv` entry for the management server are available, make sure the `root` is not registered in `/etc/ftpusers` on the managed node.

- User IDs and group IDs:

  For consistency, make sure that the user ID "`opc_op`" and the group ID "`opc_grp`" are identical on all your managed nodes.

- NIS clients:

  If the managed node is a Network Information Service (NIS or NIS+) client, you must add the HPOM default operator `opc_op` on the NIS server before installing the HPOM-agent software on a managed node. By doing so, you ensure that the HPOM default operator `opc_op` is used by HPOM and is consistent on all systems. Make sure that you adapt the user registration of adapted system resources accordingly.

Automatic Installation or Update of HPOM Agent Software

This section describes how to install or update HPOM agent software automatically by using the installation script.

Before You Begin

Before you install or update HPOM, you need to understand how to work with the installation script, root passwords, and managed nodes.

Installation Scripts

When you install, update, or remove HPOM agent software, you use the `inst.sh(1m)` script. If the connection between management server and managed node is lost during installation or upgrade of the HPOM agent software upgrade, the `inst.sh` script tries to reconnect automatically to the agent and returns an error if it fails.

By default, `inst.sh(1m)` uses ping to send 64-byte ICMP packets when installing the agent. If you are installing the agent through a firewall that does not allow 64-byte ICMP packets, reduce the packet size before installing the agent, for example:

```
# ovconfchg -ovrg server -ns opc -set OPC_PING_SIZE 56
```

To avoid the verbose output produced by the `ovconfchg` script, set a shell variable for user `root`, as follows:

Bourne/Korn	`OPC_SILENT=1; export OPC_SILENT`
C	`setenv OPC_SILENT`

Root Passwords

Before you can begin agent-software maintenance, you need to know either the root passwords of the managed nodes, or you must make `.rhosts` entries available for user root (UNIX only). Failing that, make sure the local `/etc/hosts.equiv` (on the UNIX managed nodes) contains an entry for the management server.

Managed Nodes

Before installing or removing HPOM agent software on the managed nodes, read the section "Agent Installation Tips" on page 35.

Caution: Make sure you have either `REXEC`, `RSH`, or `SSH` services enabled on the remote agent before you start the HPOM agent installation. Otherwise the agent installation will fail.

Adding a Managed Node to the HPOM Database

Note: Make sure that the SNMP agent is running before adding a managed node to the HPOM database.

Before you can install HPOM on a managed node, you must add the managed node by using the `opcnode` command line tool, for example:

```
# opcnode -add_node node_name=<node_name> net_type=<network_type>
mach_type=<machine_type> group_name=<group_name> node_type=<node_type>
```

For detailed information, see the *opcnode(1m)* manual page.

Automatic Software Installation and Update

Note: HPOM agent software installation does not include configuration distribution.

To install or update the HPOM agent software automatically, use the `inst.sh` script. The installation script `inst.sh(1m)` verifies that all specified systems are reachable and accessible by the `root` user. If the connection between management server and managed node is lost during the installation or upgrade of the HPOM agent software, the `inst.sh` script tries to reconnect automatically to the agent and returns an error if it fails.

Watch the script execution carefully. Your interaction might be required if any errors or warnings occur. For example, if a password is required, the script prompts you to supply one before continuing the installation process. When the script is finished, verify the overall result of the script run.

Check the local (managed node) installation log file for any problems.

If you could not review the installation process in a terminal window, check the following log file on the management server for errors or warnings:

`/var/opt/OV/log/OpC/mgmt_sv/install.log`

Running Multiple Agent Installation Instances in Parallel

By default, the `inst.sh` script does not allow running multiple instances in parallel.

To enable multiple instances of `inst.sh` to run in parallel set the configuration variable `OPC_AGT_MULTI_INST` to `TRUE`, as follows:

```
ovconfchg -ovrg server -ns opc -set OPC_AGT_MULTI_INST TRUE
```

The semaphore files are located at:

`/var/opt/OV/share/tmp/OpC/mgmt_sv/inst.lock.<node_name>`

Locking is done per node name. Therefore, running only one instance of `inst.sh` is allowed per node.

Note: It is recommended to run multiple instances of `inst.sh` in the separate terminal windows.

You can also write a wrapper script that calls multiple `inst.sh` instances by using info files. In such case, make sure that your outputs are directed properly.

Watch the script execution carefully. Your interaction might be required if any errors or warnings occur. For example, if a password is required, the script prompts you to supply one before continuing the installation process. When the script is finished, verify the overall result of the script run.

Check the local (managed node) installation log file for any problems. If you could not review the installation process in a terminal window, check the following log file on the HP Operations management server for errors or warnings:

`/var/opt/OV/log/OpC/mgmt_sv/multi_install.log`

Note: You can always fallback to single `inst.sh` instance mode by clearing the `OPC_AGT_MULTI_INST` variable:

```
ovconfchg -ovrg server -ns opc -clear OPC_AGT_MULTI_INST
```

Secure Shell Installation

This section describes how to use Secure Shell (SSH) software for installing HPOM agent software on managed nodes.

The SSH installation method provides enhanced security for installations that are performed over insecure lines (for example, over the Internet).

Note: HPOM does not provide the SSH software. If you want to use SSH for the HPOM agent installation, you must first install and configure the SSH software on the management server and the managed node.

There are two SSH protocol versions available: SSHv1 and SSHv2. The HPOM agent installation uses whichever version of the SSH protocol that is available on the management server and the managed node.

Hardware and Software Requirements

This section describes the hardware and software requirements for installing HPOM agents on the managed nodes using the SSH installation method.

For a list of managed node platforms and operating system versions on which the SSH installation method is supported, see the *HPOM Installation Guide for the Management Server*.

- Communication:

  Make sure that the SSH client and server (daemon) is installed and fully configured on both the HPOM management server and the managed nodes.

- User logons:

  Log on without a password for user `root` must be enabled on both the HPOM management server and the managed nodes. See "Installing HP Operations Agent Software by Using SSH" on page 42.

  Note: Logon without a password is only required during the initial installation of the HPOM agent. You can disable it afterwards. Subsequent upgrades to the agent software are handled by the BBC Local Location Broker.

Installing HP Operations Agent Software by Using SSH

To install HP Operations agent software by using the SSH installation method, follow these steps:

1. Configure logon for user root.

   The recommended method to configure logon without a password is RSA authentication, based on the user's public/private key pair and the `ssh` agent utility.

   To configure a logon using the provided utilities, follow these steps:

   a. If you are setting up an HP-UX managed node, make sure that the `sshd` configuration options in `/usr/local/etc/sshd_config` are set as follows:

   ```
   AllowTcpForwarding yes
   X11Forwarding yes
   X11DisplayOffset 10
   X11UseLocalhost no
   ```

   b. Generate the key pair using the `ssh-keygen` command, as follows:

   ```
   # ssh-keygen
   ```

   ```
   Initializing random number generator...
   Generating p: ............++ (distance 186)
   Generating q: ....................................++ (distance 498)
   Computing the keys...
   Testing the keys...
   Key generation complete.
   Enter file in which to save the key (/home/username/.ssh/identity): <press
   Enter>
   ```

   Note: Make sure not to provide a passphrase. This way, no private key is needed when establishing a connection.

   ```
   Enter passphrase: <press Enter>
   Enter the same passphrase again: <press Enter>
   Identification has been saved in /home/username/.ssh/identity.
   Your public key is:
   1024 35 718535638573954[...] username@local

   Public key has been saved in /home/username/.ssh/identity.pub
   ```

   c. Use `ssh` to connect to the managed node, and from there connect back to the management server.

   This step creates the `$HOME/.ssh` directory on the managed node, as well as some files in that directory. After the directory is created, log out from the managed node.

   d. Copy the local public key to the managed node using one of the following methods:

   - Use the secure-copy command, `scp`:

     ```
     # scp .ssh/identity.pub <user>@<managednode>:.ssh/authorized_keys
     ```

   - Use the secure-shell command, `scp`:

```
# ssh <user>@<managednode> 'cat >> ~/.ssh/authorized_keys'
<~/.ssh/identity.pub
```

Note: Because the `~/.ssh/authorized_keys` file can contain many keys, it is important that it is not overwritten during the preparations for the installation on a new system. The second method for transferring public key mentioned above, will not overwrite the file.

e. During the HPOM agent installation, `ssh` and `scp` executables must reside in one of the following recommended locations:

- `/usr/bin/`
- `/usr/sbin/`

Create a soft link to the `ssh` executable. For example:

```
# ln -s /usr/local/bin/ssh /usr/bin/ssh
# ln -s /usr/local/bin/scp /usr/bin/scp
# ln -s /usr/local/sbin/sshd /usr/sbin/sshd
```

2. Set up managed nodes for HPOM agent installation using SSH.

When the `inst.sh` script prompts you to enter the distribution method for the agent package, choose `4=Secure Shell installation (default=1)`.

Installing HP Operations Agent Software by Using SSH Agent Installation Method

This section describes how to use the Secure Shell (SSH) agent to install the HP Operations agent software on managed nodes. The difference between the method described in this section and the method described in "Installing HP Operations Agent Software by Using SSH" on page 42 is that, for the agent software-installation method described in this section, you must provide a password before the installation starts. The agent software-installation method described in this section helps prevent password-less logons to managed nodes by the root user from the HP Operations management server.

To install the HPOM agent software using the secure shell method, perform the following steps:

1. Generate and distribute a password-protected key (identity):
   a. Run `ssh-keygen` as described in "Installing HP Operations Agent Software by Using SSH" on page 42.

   Caution: When prompted, make sure that you provide a password.

   b. Distribute keys to the managed nodes as described in "Installing HP Operations Agent Software by Using SSH" on page 42.
2. Run the SSH agent and set environment variables that are required by the SSH agent:

```
# eval `ssh-agent`
```

You can do it also manually by first running the SSH agent, and then the commands, which the SSH agent lists:

```
# ssh-agent SSH_AUTH_SOCK=/tmp/ssh-fbdkZc4730/agent.<pid>;
export SSH_AUTH_SOCK;SSH_AGENT_PID=<pid>;
export SSH_AGENT_PID;
echo Agent pid <pid>;
```

3. Add the key to the SSH agent database and, when prompted, enter the password you created in the previous step:

   ```
   # ssh-add <identity_file_name>
   ```

   For example:

   ```
   # ssh-add /home/username/.ssh/identity
   ```

   Note: The SSH agent imports all keys under `/home/username/.ssh/` if it is run without the arguments.

4. Run the SSH agent installation, as described in "Installing HP Operations Agent Software by Using SSH" on page 42.
5. Remove the key from the database, or stop the SSH agent by running the following command:

   ```
   # ssh-add -d <identity_file_name>
   ```

Agent Bootstrap Installation

Agent bootstrap installation uses secure file transfer and execution methods based on the SSH communication protocol. For this purpose HPOM uses PuTTY, which is a third-party utility that serves as an SSH client. For using this utility it is required that an SSH server is running on a remote node.

Agent bootstrap installation brings on the following benefits:

- passwordless SSH logon configuration is no longer required
- enabling `rexec`, `rsh`, and `ftp` services on a remote node is no longer required
- passwords are obfuscated
- managed nodes do not require additional configuration after the operating system restore or reinstallation (only SSH daemon must be run)

Caution: PuTTY is not HP software. It is provided “as is” for your convenience. You assume the entire risk relating to the use or performance of PuTTY.

Overview

Agent bootstrap installation breaks the remote installation of an agent into the following steps:

1. Transferring the packages
2. Starting the install commands (for example, `oasetup.sh`)

During bootstrap, LCoreDeploy component uses the mechanisms configured to transfer the packages to the remote node (for example, `pscp` PuTTY utility) and the mechanisms for running the installation commands (`plink` PuTTY utility). However, if an agent is already installed on the node it uses its own mechanisms.

All the files that are part of the bundle `.xml` are transferred to the *<OvDataDir>*`/installation/incoming/`*<bundle name>* directory on the managed node. These files are copied to the remote node by using the `COPY` command as defined in the mechanism's configuration settings.

Both the mechanisms for copying the files and for running the commands represent one or two non-interactive executables that use remote node names, user names, passwords, source files, target files, and commands as command-line parameters to run.

The agent bundle `.xml` file (`OVO-Agent.xml`) is parsed to identify all the installation commands such as `preinst`, `exeinst`, and `postinst`. To run each of the commands, agent bootstrap uses the `EXEC` command as defined in the mechanism's configuration settings.

By default, agent bootstrap uses PuTTY utilities provided with the HP Operations management server. To check default configuration settings for the file transfer and execution methods (`COPY` and `EXEC` commands) on the HP Operations management server, run the following commands:

Caution: It is not recommended to change the following settings.

```
ovconfget -ovrg server depl.mechanisms.hpomssh
```

```
COPY=/opt/OV/bin/OpC/agtinstall/runpscp.sh <host> <sourcefile> <targetfile> <user> <passwd>
```

```
EXEC=/opt/OV/bin/OpC/agtinstall/runplink.sh <passwd> <user> <host> <command>
```

Note: You can also use `ssh` provided by the operating system for the file transfer and execution methods. However, you must configure the passwordless logon for such a setup.

Running Agent Bootstrap Installation

Agent bootstrap installation is initiated by the `inst.sh` script.

Note: Agent bootstrap installation is now default installation method (value=5).

For example, after running `inst.sh` in the interactive mode the following question is displayed:

```
========> Enter distribution method for Agent package (1 = Management Server based,
3 = Windows NT installation server 4 = Secure Shell installation, 5 = Agent
Bootstrap  (default=5)?
```

Note: If you prefer using `info_file` for running the `inst.sh` script make sure that you properly specify the installation method (parameter No. 12). For details see the *inst.sh* manual page.

Setting the Bootstrap Installation Method for Nodes Added by Using the Administration UI

When adding managed nodes by using the Administration UI, you can specify your preferred agent installation method for each of the nodes. However, if you have a larger number of nodes, and you want to use the bootstrap installation method for all of them, you can avoid selecting this method for the each node respectively by setting this method as a default on the HP Operations management server. To do so, follow these steps:

1. In the command line editor, type the following command:

   ```
   ovconfchg -ovrg server -ns opc -set OPC_USE_NEW_BOOTSTRAP_METHOD TRUE
   ```

2. Refresh the Administration UI with the newly specified setting by typing the following:

   ```
   /opt/OV/bin/OpC/opcagtdbcfg -p <platform_selector> -dbupdate -force -show |
   grep ACCESS_METHOD
   ```

   In this instance, *<platform_selector>* is an operating system of the nodes you want to add.

Note: You can set the default bootstrap installation method in this way only for the newly added nodes. For the nodes that already exist in the Node Bank, the preferred installation method can be changed only manually on the Edit Node page in the Administration UI.

HPOM Software Removal on the Managed Node

To remove the HPOM agent software from the managed node, follow these steps:

1. Stop all HPOM agent software running on the managed node.
2. Remove the agent software from the managed node.

For more information about the platform-specific commands you must enter during the software-removal process, see the HP Operations agent documentation.

Note: Multiple instances of `inst.sh` to running in parallel support also HPOM software removal on

the managed node. To enable multiple instances of `inst.sh` to run in parallel set the configuration variable `OPC_AGT_MULTI_INST` to `TRUE`, as follows:

```
ovconfchg -ovrg server -ns opc -set OPC_AGT_MULTI_INST TRUE
```

If the `inst.sh` script was not used to remove the agent software, you must perform some additional actions on the management server after removing the HPOM agent software from a managed node, as described in "Cleaning Up After Removing a Managed Node from HPOM" on page 48.

Cleaning Up After Removing a Managed Node from HPOM

To manually clean up after removing the HPOM agent software from a managed node, perform the following steps:

1. Update the HPOM database to reflect the removal of the agent software from the managed node. Use the `opcsw` command as follows:

   ```
   # opcsw -de_installed <node_name>
   ```

2. Remove references to the managed node from the HPOM Node Bank and the HPOM database. Use the `opcnode` command as follows:

   ```
   # opcnode -del_node node_name=<node_name> net_type=<network_type>
   ```

   In this instance, *<node_name>* is the name of the managed node that you want to remove from the HPOM database and *<network_type>* is the type of managed node (for example, Non IP, IP (Network), or External (Node)).

   The `opcnode` command also ensures that the managed node's assignment to any node groups is removed. For more information about the `opcnode` command and its parameters and options, see the *opcnode(1m)* manual page.

HPOM Agent Software Management

Frequently, managed nodes, including those with the same architecture, do not run the same operating system versions. Different operating systems are used for different purposes, for example:

- Production systems:

  Run approved operating systems versions where all required applications are available.

- Development systems:

  Run the approved or latest operating systems versions.

- Test systems:

  Run approved or latest operating system versions.

Displaying Versions of Available Agent Packages

To display a summary of all HPOM agent packages including the supported operating-system versions that are currently available on the management server, run the following script on the management server:

```
# /opt/OV/bin/OpC/agtinstall/opcversion -a
```

The latest possible HPOM agent version supporting the operating system version of the managed node is probably installed on that node. See "Displaying Versions of Installed Agent Packages" on page 49 for information about how to query the version of the installed agent software.

The related HP Operations agent software for each supported architecture is available in the following directory:

```
/var/opt/OV/share/databases/OpC/mgd_node/vendor/<platform_selector>/<hpom_
version>/RPC_BBC
```

In this instance, *<platform_selector>* is a specific platform and *<hpom_version>* is the version of HPOM that supports the specified agent platform.

Displaying Versions of Installed Agent Packages

To display the version number of the HPOM agent software that is currently installed on a managed node, run the following command on the management server:

```
# /opt/OV/bin/OpC/opcragt -agent_version <node>...
```

The `opcragt` command returns more than a single version number; it lists the individual HP software component installed on the managed node.

For more information about possible restrictions of this command, see the *opcragt(1m)* manual page .

Managed Node Administration with Subagent ID Values

If the managed-node communication type is HTTPS, the `opcragt` command with the `-id` parameter enables you to specify the subagent ID either as a number or a name. For more information about the `-id` option and the `opcragt` command, see the command's manual page.

If the `-id` is specified as a name, HPOM can communicate with the selected node directly. If the `subagent id` is specified as a number, the number must be mapped to a `subagent id name` in the `subagt_aliases` file, which is located in the following directory:

```
/etc/opt/OV/share/conf/OpC/mgmt_sv/
```

The following mappings are defined by default in the `subagent_aliases` file:

0	AGENT
1	EA
12	CODA

If a mapping between number and name is required but is not defined, the `opcragt` command displays the following error message:

```
Subagent XXX:
Subagent not registered.
```

Note: The `-id` option is provided for backward compatibility with older versions of HPOM.

For example, you can use the `-id` option to specify the subagent ID when querying the status of the subagent processes on the managed node or when stopping and starting the subagent processes.

Querying the HPOM Subagent Status

To display the current status of the HPOM subagent using the subagent ID, use the `opcragt` command with the `-start` and `-id` options, as follows:

```
# opcragt -id CODA <node_name>
```

```
Node <node_name>:
HPOM Managed Node status:
-------------------------
OV Control                ovcd                (12338) is running
OV Config and Deploy      ovconfd             (12342) is running
OV Performance Core       coda                (12345) is running
OV Communication Broker   ovbbccb             (12339) is running
Done.
```

Stopping and Starting HPOM Subagents

To use the subagent ID to start or stop the HPOM subagent on a managed node, use the `opcragt` command with the `-start` and `-id` options, as follows:

```
# opcragt -start -id CODA <node_name>
```

```
Node <node_name>:
Starting OpC services...Done.
```

In this instance, *<node_name>* is the name of the managed node where you want to use the subagent ID to stop or start the HPOM subagent.

Subagent Management in HPOM

Subagents are components that are not a part of the default HPOM distribution but can be partially managed from HPOM. Some of the subagents are controlled by the OV Control daemon.

Caution: The term “subagent” is used in several contexts to mean different things. In this section, subagent refers to third-party software which runs as a subagent and can be controlled by HPOM. However, “subagent” is also used to refer to parts of the agent that can be individually started and stopped, for example, using the `opcragt` command, as described in "Managed Node Administration with Subagent ID Values" on page 49.

Prerequisites for Managing Subagents

To manage subagents in HPOM, you need to understand the underlying concepts and be aware of the prerequisites set by the provider of the subagent software. For information about subagent concepts, see the *HPOM Concepts Guide*. For more information about the management and administration tasks for HPOM subagents, see "Subagent Administration in HPOM" on page 51, which includes details of subagent assignment, installation, and activation.

Subagent Administration in HPOM

When you install the subagent software packages on the HPOM management server, there are some tasks you should perform to ensure that subagents are properly installed and functioning on managed nodes. The tasks are outlined in the following topics:

- "Subagent Assignment to Managed Nodes" on page 51
- "Subagent Installation on Managed Nodes" on page 52

To avoid problems during the distribution of subagent to managed nodes, consider also the tasks presented in the following topics:

- "Activating the Subagent" on page 53
- "Resolving Migration Problems" on page 53

Subagent Assignment to Managed Nodes

Subagents are assigned to managed nodes in the same way that their corresponding subagent-registration policies are assigned to the nodes. If assigned policies are placed in the appropriate policy group, too, both the subagent and its configuration are assigned simultaneously when the policy group is assigned to a node.

Note: If the appropriate node type is not present in the subagent registration file, the installation of the subagent fails.

For information about assigning policies and policy groups to managed nodes, see the *HPOM Concepts Guide*.

Subagent Installation on Managed Nodes

The `opcbbcdist` process uses already prepared distribution description files to install the subagent on the managed node. These files are placed upon the subagent software installation at the predefined location on an HPOM management server.

Installing the Subagent Software

To install the subagent software on the managed node, type the following:

```
# opcragt -subagent -install <subagent_name> <node_name>
```

In this instance, the *<node_name>* is the name of the node on which you want to install the subagent.

Removing the Subagent Software

To uninstall the subagent software, use the `opcragt` command with the following options:

```
# opcragt -subagent -uninstall <subagent_name> <node_name>
```

Installing all Subagents

To install all assigned subagents on a managed node simultaneously, use the `opcragt` command with the following options:

```
# opcragt -distrib -subagts <node_name>
```

Redistributing the Subagent Software

To redistribute the subagent software, use the `opcragt` command with the following options:

```
# opcragt -subagent -reinstall <subagent_name> <node_name>
```

Note: Using the `opcragt` command with the `-force` option does not trigger the redistribution process. This is to prevent the unnecessary deployment of subagent packages on the agent. For more information about the `opcragt` command options, see the *opcragt(1m)* manual page.

To find out how to configure installed subagent packages, see the documentation supplied with the subagent packages.

Activating the Subagent

To activate the subagent on a particular node, use the `opcragt` command with the following options:

```
# opcragt -subagent -active <subagent_name> <node_name>
```

Activating a subagent means setting the active flag for this subagent in the HPOM database.

Because the active flag indicates that the subagent is already installed on the managed node, this subagent will not be installed again on this particular managed node during the subagent distribution process.

Activating a subagent is useful when an agent was either manually installed on the managed node, or it was installed from another HPOM management server. When the configuration is migrated from one HPOM server to another, it is especially advisable to activate subagents for managed nodes on the target HPOM server. In this case, the subagent packages may not be transferred as well, and if the subagents are not activated, error messages appear upon subsequent distribution.

Resolving Migration Problems

To avoid problems during the distribution of subagent software to the managed nodes, perform one of the following tasks:

- Install all subagent packages on the HPOM management server where you intend to upload the downloaded configuration.
- Remove subagent registration policies from the policy groups when they are uploaded (or deassigned from managed nodes, if they are previously directly assigned). Note that this makes it impossible to obtain a complete inventory report for particular subagents on managed nodes.

If you migrate the configuration from one HPOM server to another, subagent packages are not downloaded along with the policies. This results in failure of the subsequent subagent distribution to nodes, since subagent registration policies point to non-existing subagent packages. To avoid error messages during subsequent distributions, activate the subagents for managed nodes on the target HPOM server. For more information, see "Activating the Subagent" on page 53.

Software Installation and Removal on Managed Nodes

HPOM provides tools that help debug the installation and removal of the HPOM agent software on the managed nodes. These tools help developers when testing HPOM installation scripts for new platforms, and assist users in examining errors that occur during the installation of the HPOM agent software.

Debugging Tools for Software Installation and Removal

The following tools are available to help debug problems that occur when installing or removing HPOM agent software on the managed node:

- Command tracing:

  Prints shell commands and their arguments from installation programs into a file specified in the file `inst_debug.conf` as argument of the environment variable `OPC_DEBUG_FILE`.

- Event tracing:

  Can be used in addition to command tracing to record important events of the installation process into the existing installation log file:

  `/var/opt/OV/log/OpC/mgmt_sv/install.log`

You can debug the installation or removal process locally (on the management server) and remotely (on the managed node). A debug definition file `inst_debug.conf` is provided to force debugging and to specify debug options.

Enabling Debugging

You can trace what happens during the installation and removal of the HPOM agent software. The `inst_debug.conf` file must be edited before the agent software-installation process starts. The default `inst_debug.conf` template is located in `/etc/opt/OV/share/conf/OpC/mgmt_sv/` and can only be edited by user `root`.

To enable debugging of the installation and removal of the HPOM agent software, follow these steps:

1. Make a copy of the default `inst_debug.conf` file and place it in the directory `/var/opt/OV/share/tmp/OpC/mgmt_sv` by using the following command:

   ```
   # cp /etc/opt/OV/share/conf/OpC/mgmt_sv/inst_debug.conf
   /var/opt/OV/share/conf/OpC/mgmt_sv/inst_debug.conf
   ```

2. Specify the parts of the agent software-installation process that you want to trace and debug by editing the copy of the `inst_debug.conf` file you made in the previous step.

   a. Enable the environment variables you want to use by removing the leading comment (#), for example:

      `#OPC_DEBUG_FILE=/tmp/inst.sh.2`

   b. Change the variable value to suit the demands of your environment. For example, you can specify the name of the file that you want to use to store the debugging information produced during the installation or removal of agent software, as follows:

      `OPC_DEBUG_FILE=/tmp/install_debug.log`

3. Save the modified copy of the `inst_debug.conf` file.

Note: The syntax of the file `inst_debug.conf` is not checked at any time, either during modification or when you save it. If the `inst_debug.conf` file contains any syntax errors, the installation or removal process aborts.

For a detailed description of the (de-)installation debug facilities, as well as examples of the `inst_debug.conf` file, see the *inst_debug(5)* manual page.

Disabling Debugging

To disable debugging, remove (or rename) the copy of the `inst_debug.conf` file that you copied to the following location when enabling the debugging feature for agent software installation and removal, as described in "Enabling Debugging" on page 54.

To disable debugging of the installation and removal of the HPOM agent software, perform the following steps:

1. Locate the copy of the `inst_debug.conf` file that you used to enable the debugging of agent software installation:

   ```
   # cd /var/opt/OV/share/conf/OpC/mgmt_sv
   ```

2. Rename the copy of the `inst_debug.conf` file that you used to enable the debugging of agent software installation:

   ```
   # mv inst_debug.conf inst_debug.conf.TMP
   ```

Chapter 2: HPOM Configuration

In this Chapter

This chapter describes the preconfigured elements provided with HP Operations Manager (HPOM). It also describes how to integrate applications into HPOM. To better understand the elements and windows you can use to customize these preconfigured elements, see the *HPOM Concepts Guide*.

The information in this section covers the following topics:

Preconfigured Elements

This section describes defaults for managed nodes, message groups, and message ownership.

By default, the management server is configured as a managed node with the default policies for SNMP event interception, HPOM message interception, log-file encapsulation and monitoring.

Default Node Groups

HPOM provides default node groups for the management server. You can add, modify, delete, and hide these default node groups, as needed.

Managing Node Groups

You can add, modify, delete, and list node groups by using the `opcnode` command line tool HPOM. For more information, see the *opcnode(1m)* manual page.

Assigning Categories to Node Groups

Categories can also be assigned to node groups. Categories unify the related instrumentation files and make their distribution to the node groups easier. For more details, see "Category-based Distribution of Instrumentation" on page 215.

Default Message Groups

HPOM provides default message groups. You can add, review, and delete message groups. For more information, see the *HPOM Administration UI Help*.

Details about individual message groups provided with HPOM are shown in Table 1.

Table 1: HPOM Default Message Groups

Message Group	Description
Backup	Messages about backing up and restoring HPOM (for example, `fbackup(1)`, HP Data Protector, HP OmniStorage, Turbo-Store).
Certificate	Messages relating to certificate handling.
Database	Messages about database problems
HA	Messages about high-availability problems.
Hardware	Messages about hardware problems
Job	Messages about job streaming.
Misc	Messages that are not assigned to any other message group. If a message does not have a message group assigned, or if the message group is not configured, the message automatically belongs to the Misc message group. It is not permitted to delete or rename the Misc message group.
Network	Messages about network or connectivity problems.
NNMi	Messages concerning incidents forwarded from the Network Node Manager i product(s).
OMU Admin UI	Messages concerning problems with the administrator's graphical user interface.
OpC	Messages generated by HPOM itself. This message group should not be used by `opcmsg(1|3)`. It is not permitted to delete or rename the OpC message group.
OpenView	Messages generated by HPOM itself. This message group should not be used by `opcmsg(1|3)`. It is not permitted to delete or rename the OpenView message group.

HPOM Default Message Groups, continued

Message Group	Description
OS	Messages about malfunctions in the operating system, I/O, and so on.
Output	Messages about print spooling and hardcopy functionality, for example: `lp(1)`, `lpr(1)`.
Performance	Messages about hardware malfunctions (that is, CPU, disk, or process malfunctions) and software malfunctions (for example, HP Performance malfunctions).
Security	Messages about security violations or attempts to break into a system.
SiS Monitoring	Messages originating from HP SiteScope and forwarded by the HP SiteScope Adapter.
SNMP	Messages generated by SNMP traps.

Managing Message Groups

You can add, modify, delete, list message groups and change message group attribute values by using the `opcmsggrp` command line tool HPOM. For more information, see the *opcmsggrp(1m)* manual page.

Message Ownership

HPOM message ownership enables users to mark or own messages. The information in this section explains what marking and owning means and how you can configure HPOM to indicate who owns a message and how ownership is displayed.

By marking or owning a message, you can inform others that you have taken responsibility for resolving the message's underlying problem. Marking or owning a message restricts access to the message:

- Marking a message:

  Operator or administrator takes note of a message.

- Owning a message:

  Operator or administrator either chooses to take charge of a message or is forced to take charge of a message, depending on how your environment has been configured. The operator or administrator must take charge of the message to carry out actions associated with that message.

HPOM enables you to configure the way in which message ownership is displayed and enforced. To display message ownership, you set the message-ownership display mode; to enforce message ownership, you set the message-ownership mode.

The information in this section covers the following topics:

Ownership-Mode Types

You set the ownership policy by selecting one of the following default ownership modes:

Optional	User may take ownership of a message. Use the option `OPC_OWN_MODE OPTIONAL`, as described in "Optional Ownership Mode" on page 59.
Enforced	User must take ownership of messages. Use the option `OPC_OWN_MODE ENFORCED`, as described in "Enforced Ownership Mode" on page 59.
Informational	Concept of ownership is replaced with that of marking messages. A marked message indicates that an operator has taken note of a message. Use the option `OPC_OWN_MODE INFORM`, as described in "Informational Ownership Mode" on page 60.

Optional Ownership Mode

In optional ownership mode, the owner of a message has exclusive read-write access to the message. All other users who can view the message in their browsers have only limited access to it.

In optional ownership mode, only the owner of a message may perform the following operations:

- Message actions:

  Perform operator-initiated actions related to the message.
- Message acknowledgement:

  Acknowledge the message (that is, move the message to the history database).

Enforced Ownership Mode

In enforced ownership mode, a user either chooses explicitly to take ownership of a message, or is assigned the message automatically. A message can be assigned to an operator if the operator attempts to perform operations on a message that is not owned by any other operator.

In enforced mode, ownership of a message is assigned to any operator who performs any of the following operations on a message:

- Message actions:

  Perform an operator-initiated action attached to the message.
- Message unacknowledgement:

  Unacknowledge a message, that is: move the message from the history browser to the active-messages browser.

Informational Ownership Mode

In informational mode, a marked message indicates that an operator has taken note of a message. Marking a message is for informational purposes only. Unlike optional and enforced modes of message ownership, informational mode does not restrict or alter operations on the message. Note that operators can unmark only those messages they themselves have marked.

Configuring Message-Ownership Mode

To specify the message-ownership mode, use the `ovconfig` command with the following options:

```
# ovconfchg -ovrg server -ns opc -set OPC_OWN_MODE <ownership_mode_value>
```

Where *<ownership_mode_value>* is one of the following:

- `ENFORCED`
- `OPTIONAL`
- `INFORM`

If the ownership mode is not specified, HPOM assumes the default value `OPC_OWN_MODE ENFORCED`. For more information about the different message-ownership modes, see "Ownership-Mode Types" on page 59.

Ownership Display-Mode Types

HPOM provides different ways to configure the way in which message ownership is displayed and enforced. HPOM provides the following ownership-display modes:

- No status propagation:

  Default setting: Uses the option `OPC_OWN_DISPLAY NO_STATUS_PROPAGATE`. For more information, see "No-Status-Propagation Display Mode" on page 61.
- Status propagation:

  Uses the option `OPC_OWN_DISPLAY STATUS_PROPAGATE`. For more information, see "Status-Propagation Display Mode" on page 61.

No-Status-Propagation Display Mode

If the display mode is set to NO_STATUS_PROPAGATE, the severity color of a message changes when the message is owned or marked.

HPOM uses the following default colors to indicate ownership:

Pink	Message is owned by you.
Beige	Message is owned by someone else.

In addition, the own-state color bar at the bottom of the Java GUI Message Browser reflects the new number of messages owned.

Status-Propagation Display Mode

If the ownership-display mode is set to STATUS_PROPAGATE, the status of all messages (whether they are owned or not) is used for the propagation of severity status to the related symbols of other submap windows. In this display mode, the only indication that the message is owned, is a flag in the own-state column in the Java GUI Message Browser.

For information on how to configure the ownership and ownership-display modes, see "Configuring Message-Ownership Mode" on page 60.

Changing Ownership Display Modes

To change the ownership display mode, perform the following steps:

1. Change the ownership display mode using the ovconfchg command with the following parameters and options:

   ```
   # ovconfchg -ovrg server -ns opc -set OPC_OWN_DISPLAY <ownership_display_mode_value>
   ```

   Note that you can set the *<ownership_display_mode_value>* to one of the following values:

   - STATUS_PROPAGATE
   - NO_STATUS_PROPAGATE

   If the ownership display mode is not specified, HPOM assumes the default value NO_STATUS_PROPAGATE. For more information about message-ownership display modes, see "Ownership Display-Mode Types" on page 60.
2. Reload the configuration of any connected Java GUI. For more information about the configuration of the Java GUI, see the *HPOM Java GUI Operator's Guide.*

Policy Groups

You can use the `opcpolicy` command line tool to add, modify, or delete policies and policy groups. For more information about command options, see the *opcpolicy(1m)* manual page.

Note: There can be multiple versions of policy groups on managed nodes. For information about managing these versions of policy groups, see "Managing Policy Groups Versions" on page 148.

Default Policy Groups

HPOM provides a variety of policy groups that you can use for the monitoring and configuration of the HP Operations management server. You can use the `opcpolicy` command to list the installed policy groups and check the contents of individual policy groups. The `opcpolicy` command is located in the `/opt/OV/bin/OpC/utils` directory.

The following default policy groups are provided with the HPOM management server:

- Correlation Composer
- Examples:
  - ECS
  - Unix
  - Windows
- Management Server
- SNMP
- SiteScope Integration/*<SiteScope Policy Group>*

Displaying a List of Policy Groups

To display a list of the policy groups deployed on an HPOM managed node or the HPOM management server, use the `opcpolicy` command with the `-list_groups` option, as follows:

```
# /opt/OV/bin/OpC/utils/opcpolicy -list_groups
```

The `opcpolicy` command with the `-list_groups` parameter displays the following information on the HPOM management server:

```
[root@omlinux1 ~]# opcpolicy -list_groups
---------------------------
policy group: /Correlation Composer
---------------------------
policy group: /Examples
---------------------------
```

```
policy group: /Examples/ECS
---------------------------
policy group: /Examples/Unix
---------------------------
policy group: /Examples/Windows
---------------------------
policy group: /Management Server
---------------------------
policy group: /SNMP
---------------------------
policy group: /SiteScope Integration
...
---------------------------
```

Displaying the Contents of a Policy Group

To display a list of the policies contained in an individual policy group, use the `opcpolicy` command with the `-list_group` option as follows:

```
# /opt/OV/bin/OpC/utils/opcpolicy -list_group "group=<PolicyGroup_Name>"
```

<PolicyGroup_Name>	Name of the HPOM policy groups whose contents you want to list, for example "Management Server".

The example of the `opcpolicy` command displays the following output:

```
---------------------------
policy group: /Management Server
assigned policy    : opcmsg(1|3), version 0009.0000, LATEST
assigned policy    : distrib_mon, version 0009.0000, LATEST
assigned policy    : mondbfile, version 0009.0000, LATEST
assigned node      : omlinux1
---------------------------
```

Default Users

HPOM provides a number of user configurations. You can customize these default settings to match the specific requirements of your organization.

Default User Types

Standard HPOM user configurations include the following:

opc_adm	HPOM administrator
opc_op	HPOM operator

Note: The default HPOM user opc_op is not the same as the user account opc_op created on all UNIX managed nodes during the installation of the HPOM agent. The default home directory of the HPOM agent user account opc_op is /home/opc_op on HP-UX and /export/home/opc_op on Solaris.

Default HPOM User Names and Passwords

For a list of default user names and passwords for the preconfigured users, see Table 2.

Table 2: HPOM User Names and Passwords

Default User	Default User Name	Default Password
HPOM administrator	opc_adm	OpC_adm
HPOM operator	opc_op	OpC_op

HPOM Administrators

With the Administration UI, HPOM supports the configuration and use of multiple concurrent HPOM administrators, whose responsibility it is to set up and maintain the HPOM software. During configuration of the HPOM administrator, you can specify very fine levels of access to either individual objects or object groups (for example, R (read), C (create), M (modify), D (delete), A (assign), and E (execute)).

Note that it is not permitted to modify the HPOM administrator's logon name, opc_adm. Note, too, that the opc_adm user is permitted by default to access the HPOM server application-programing interface (API). For more detailed information about configuring HPOM users using the command-line interface, see the *opccfguser(1m)* manual page.

Default Operator Types

HPOM provides the opc_op default operator. This operator is preconfigured and can be used as a basis for creating new operator profiles that more accurately reflect the needs of a given organization or environment. For more information on the HPOM operators, see the *HPOM Concepts Guide*.

Default Node Group Types

The following node groups are assigned to the opc_op operator by default:

- hp-ux
- Linux
- Solaris

Default Message Group Types

The following message groups are assigned to the `opc_op` operator by default:

- Backup
- Database
- HA
- Hardware
- Job
- Misc
- Network
- OpC
- OpenView
- OS
- Output
- Performance
- Security
- SNMP

The messages each operator receives and the nodes those messages come from are not necessarily the same. The responsibility matrix you chose for a given operator determines which node group sends which messages to which operator.

For example, by default, all HPOM operators have the `Network` message group in their Java GUI `Object Pane`. However, the node groups that send messages associated with the `Network` message group vary according to the operator. The origin of the messages depends on the selection you make in a given operator's responsibility matrix.

Default Application Types

The applications and application groups assigned by default to the HPOM users reflect the responsibilities you assign to them.

HPOM allows you to add, delete, and move applications using the `opcappl` command-line tool. You can use the default settings as a base for configuring users and responsibilities that match the needs of individual environments. For more information about managing applications from the command-line interface, see the *opcappl(1m)* manual page.

The following applications are assigned to the `opc_op` operator by default:

- Broadcast
- HPOM Status

Default Applications and Application Groups

Default applications and application groups are provided with the HPOM server installation.

Note: HPOM applications are available for reference but no longer as default for the specified agent platforms.

Table 3 shows the default applications and application groups provided by HPOM.

Table 3: Default Tools and Tool Groups

Name	Applications	Application Group
Certificate Tools		√
NNMi		√
NNMi Int-Admin		√
Windows Tools		√
OM License Tools		√
HP Composer		√
UN*X Tools		√
Broadcast	√	
HPOM Status	√	

Listing the Application Groups Assigned to an Operator

You can use the `opccfguser` command to list the application groups that are currently assigned to an HPOM user, for example:

```
# opccfguser -list_assign_appgrp_user opc_op
Application groups of user: opc_op
Certificate Tools
```

Listing the Applications Assigned to an Operator

You can use the `opccfguser` command to list the *applications* that are currently assigned to an HPOM user, for example:

```
# opccfguser -list_assign_app_user opc_op
Applications of user: opc_op
HPOM Status
Broadcast
```

Note: The `opccfguser` command with the `-list_assign_app_user` option does not display applications assigned to a user by inheritance (for example, from an application group):

```
# opccfguser -list_assign_app_user opc_op
User opc_top does not have assigned applications
```

Note that the same is true for policies assigned to a node by indirect means (for example, because the policies are in a policy group). The `opccfguser` command displays a list of assigned policy groups but does not display the policies *contained* in the assigned policy groups, even if you explicitly ask for a list of assigned policies.

Broadcast Application

The `Broadcast` application enables you to issue the same command on multiple systems in parallel:

- UNIX:

Default user:	`opc_op`
Default password:	None required because the application is started by the HPOM action agent.

- Windows:

Default user:	`system`
Default password:	None required because the application is started by the HPOM action agent.

Note: For both UNIX and Windows operating systems, if the default user credentials have been changed by the operator, you must supply a password.

HPOM Status Application

The `HPOM Status` application issues the `opcragt` command. This application enables you to remotely generate a current status report about all HPOM agents on all managed nodes.

The HPOM Control Agent must always run on the managed nodes. Otherwise, the agents cannot remotely be accessed from the HP Operations management server.

Default user:	**root** (user must be **root**)
Default password:	None required because the application is started by the HPOM action agent.

Note: If the default user has been changed by the operator, you must supply a password.

Event Correlation

The runtime engine for HPOM event-correlation is available for the HP Operations management server and the HP Operations agent. See the *HPOM Installation Guide for the Management Server* for a list of platforms on which the runtime engine currently runs.

For more information about the concepts behind event correlation, as well as the way event correlation works in HPOM, see the *HPOM Concepts Guide*.

Configuring Event Correlation for HPOM

The HPOM message source policy allows you to specify which conditions generate a message and whether or not the generated message is copied or diverted to the message stream interface (MSI) from where it may be passed to and processed by the event-correlation template. To configure event correlation, perform the following steps:

1. Enable output from HPOM's internal message stream to the message stream interface (MSI) as required at either the management-server or managed-node level (or both), as follows:
   - On the HP Operations management server, enable output to the server MSI by running the following command:

     ```
     # opcsrvconfig -msi -enable
     ```

   - On the HP Operations managed node, enable output to the agent MSI using the Administration UI.

     Alternatively, you can use the `opcnode_modify()` API, though HPOM does not provide any command line tool to use with this API in the current version.

2. Set up the HPOM message source policy to ensure that the configured message conditions generate messages as intended. For each condition statement (a policy block starting with the `CONDITION` keyword), make sure that you specify the following:
   - A message-type attribute in the `CONDITION` or `SET` block of the policy body. Attributes can be any of the allowed attributes specified in the policy grammar. For more information about policy grammar, see the *HPOM Concepts Guide*.
   - The specified message-type attribute must match the corresponding attribute referenced in the Input node that starts the flow in the event-correlation circuit in which you want to process the message in question.
3. Enable the Copy/Divert to MSI option for each condition that you want to configure to generate a

message, using one of the following keywords in the SET section(s):

MPI_SV_COPY_MSG	Copy messages to the MSI and pass them to the HPOM server processes.
MPI_SV_DIVERT_MSG	Send messages to the MSI and remove them from the HPOM processing chain on the management server.
MPI_AGT_COPY_MSG	On managed nodes, copy messages to the MSI and pass them on to the HPOM server processes.
MPI_AGT_DIVERT_MSG	On managed nodes, send messages to the MSI and remove them from the HPOM processing chain on the management server.

4. Enable, if required, the logging options for each policy, by using one or more of the following keywords in the policy body, before specifying conditions:

LOGMATCHEDMSGCOND	Logs messages matching message conditions (section starting with MSGCONDITIONS).
LOGMATCHEDSUPPRESS	Logs messages matching suppress conditions (section starting with SUPPRESSCONDITIONS).
LOGUNMATCHED	Logs unmatched messages.

Log-File Encapsulation

For detailed information about encapsulating log files with the log-file encapsulator, see the *HPOM Concepts Guide*.

Log-file policies are configured to collect information from log files that are produced by standard installations, for example: syslog and cron log files on UNIX and Linux systems or application, system, and security event logs on Windows systems. If you are monitoring a non-standard installation, you should modify the policies to suit your particular needs.

Listing Default Log-File Policies

To display details of the log files that are monitored by default, use the opcpolicy command with the following parameters:

```
# opcpolicy -list_pols pol_type=LOG
```

You can edit existing (and configure new) log-file policies by modifying the policy body. The corresponding log-file policies must be configured so that the HPOM operator knows which system the

log file originated from, or the event which triggered the message. For information on policy body grammar, see the *HPOM Concepts Guide*.

HPOM Message Interception

By default, any message submitted with the `opcmsg(1)` command or the `opcmsg(3)` API is intercepted. For more information about using the command-line interface to set message-attribute defaults, logging options, and so on, see the manual pages *opcmsg(1)* and *opcmsg(3)*.

HPOM internal error messages can also be intercepted by the HPOM message interceptor.

Object Monitoring

Table 4 shows how HPOM monitors object thresholds on the management server.

Table 4: Object Thresholds on the Management Server

Object	Description	Threshold	Polling Interval
`disk_util`	Monitors disk space utilization on the root disk.	90%	10m
`distrib_mon` [a]	Monitors the software distribution process. Generates a message for each pending distribution and another message if the distribution is pending for longer than 30 minutes.	1m [b]	10m
`mondbfile` [a]	Monitors free space on disk, as well as the remaining space available for Oracle autoextend datafiles. [c]	0% [d]	10m
`swap_util`	Monitors SWAP utilization.	80%	5m

[a] Deployed by default.

[b] If a node does not fetch the assigned templates within <*$THRESHOLD*> (minutes) `opcmsg` generates a message for a pending distribution on the node.

[c] The `mondbfile` policy is not available with PostgreSQL.

[d] Threshold ensures that there is enough remaining disk space to enable the data file to be extended at least one more time (free disk space - next autoextend size > 0).

Monitoring MIB Objects from Other Communities

To monitor MIB objects from communities other than the default *public* community, use the `ovconfchg` command-line tool as follows on the managed nodes:

```
# ovconfchg -ns eaagt -set SNMP_COMMUNITY <community>
```

In this instance, `<community>` is the community for which the SNMP daemon `snmpd` is configured. If `SNMP_COMMUNITY` is not set, the default community `public` is used. To find out how to determine the configuration of `snmpd`, see the documentation supplied with the SNMP daemon.

Policies for External Interfaces

By default, no notification is configured. You can configure HPOM Notification Services by using the `opcnotiservice` command line interface. No trouble-ticket system interface is configured. You can set up one by using the `opctt` command line interface.

For more information about using the command-line interface to set up external notification and trouble-ticket services, see the *opcnotiservice(1m)* and *opctt(1m)* manual pages.

HPOM Policies

A policy is a configuration element consisting of data and meta information. Policies are deployed to managed nodes. The data- information part usually consists of a set of rules for generating the messages on the managed node to which the policy is deployed. While the data-information part is completely defined by the user, the meta information part is used for administrative tasks and is managed by the HPOM product. Each policy has a policy type, which means that its bodies conform to a specific set of rules. To learn more about policies and policy types, see the *HPOM Concepts Guide* and the *HPOM Administration UI Online Help.*

Policies can have multiple versions on the HPOM 9.xx management server, and are organized in a tree-like structure. See "Policy Versions" on page 79 and the *HPOM Concepts Guide* for more information.

To learn how to handle multiple versions of HPOM configuration (which includes, in addition to the policies, policy groups and the instrumentation data) on managed nodes, see "Managing Multiple Versions of HPOM Configuration on Managed Nodes" on page 147.

Policies can also contain category assignments. Categories unify the related instrumentation files and make their distribution to the managed nodes easier. For more details, see "Category-based Distribution of Instrumentation" on page 215.

Policy File-Naming Conventions

The names of policy files must adhere to the following rules:

- Policy header:

  `<uuid>_header.xml`

  For example, `33F23DD0-4092-11DE-8A39-0800200c9A66_header.xml`

- Policy body:

  `<uuid>_dataX`

  Where X is the body number. If a policy has only single body, this number can be omitted.

  For example, `33F23DD0-4092-11DE-8A39-0800200c9A66_data`

Adding Policies

Policies can be added to HPOM in one of the following ways:

- Direct upload of policies:

  Policies can be uploaded to the HPOM repository directly using `opcpolicy` command line tool or `opcpolicy_add()` API. Both mechanisms allow upload of single or multiple policies. If multiple policies are to be uploaded, they should be located in the same directory and follow the naming schema rules.

  An example of uploading a single policy using `opcpolicy` command line tool:

  ```
  # opcpolicy –upload file=970FF268-24FA-4f03-9E48-339E2F9A3827_header.xml
  ```

  If multiple policies are located in directory `/tmp/policies`, they can be uploaded using the following command:

  ```
  # opcpolicy -upload dir=/tmp/policies
  ```

  Any files that do not conform to the policy files naming schema will be ignored.

- Upload of policy data downloaded from another HPOM server:

  Transfer of policies from one HPOM server to another can be accomplished using `opccfgdwn` (download) and `opccfgupld` (upload) tools.

Note: Policies, which do not have condition or instruction IDs present in policy bodies, can sometimes be uploaded multiple times. Namely, during the upload, these missing IDs are generated and policy bodies get rewritten. Attempting to upload the original policy (without IDs) results in generating new IDs and effectively creating a “new” policy. These policies are successfully uploaded despite mainly having the same content in their bodies.

See the *opcpolicy(1m)* and *opccfgupld(1m)* manual pages for more information about command options.

Registering Policy Types

The policy type must be known on the HPOM server before policy of that type is registered. This is performed by using the `opcpoltype` command line tool, for example:

```
# opcpoltype -reg -xml /var/conf/poltypes.xml
```

Input for `opcpoltype` is an XML file, which describes the policy types registered on the HPOM server.

If you specify the `-dir` option, all files with the `.xml` extension in the specified directory are processed, and treated as policy type registration files.

For more information about the `opcpoltype` command options, see the *opcpoltype(1m)* manual page.

Note: A new policy type should be registered before any policy of that type is uploaded. If you attempt to upload a policy of an unknown type to the management server, an error is returned. Once the new policy type is registered, policies of that type can be uploaded and later deployed to the HPOM server.

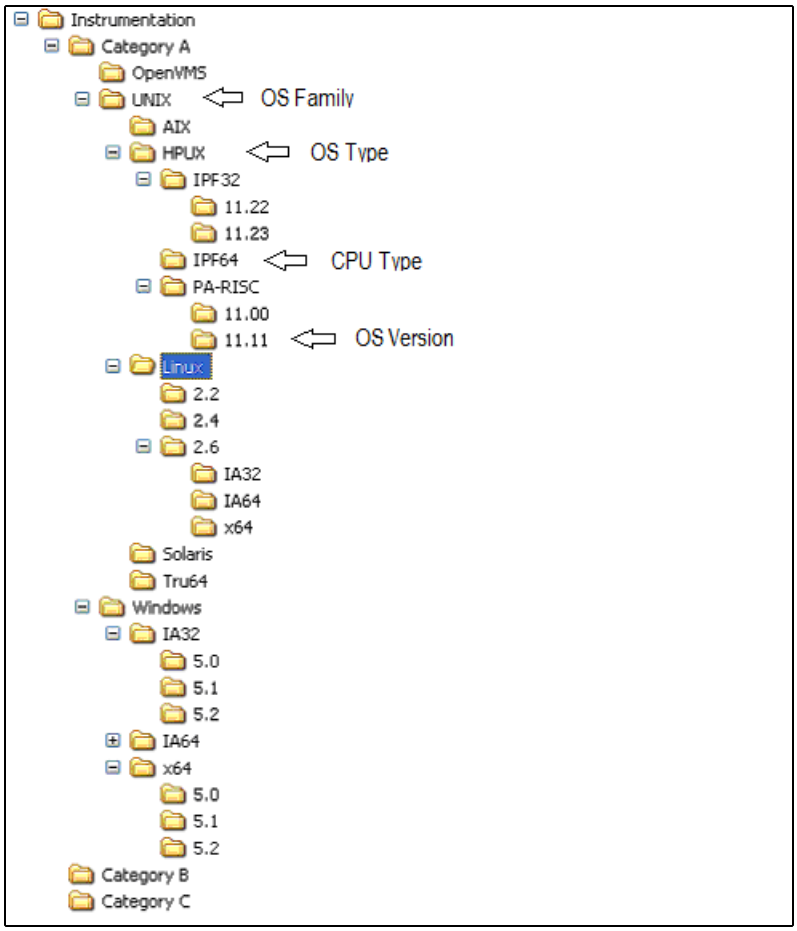
The following is an example of the XML registration file:

```
<policyTypeList>

      <policyType>
              <policyTypeName>Special-log</policyTypeName>
              <policyTypeAgentType>le</policyTypeAgentType>
             <policyTypeUUID>E8405458-2970-4DB7-825C- \
               816B3FBF11FE</policyTypeUUID>
              <policyTypeEditor>/usr/local/bin/specedit \
                -argument</policyTypeEditor>
              <policyTypeCallbacks>
                       <edit>/usr/local/bin/speccopy</edit>
                       <deploy>/usr/local/bin/specadapt</deploy>
              </policyTypeCallbacks>
             <policyTypeTemplate>/usr/local/templates/ \
               special-log.tmpl</policyTypeTemplate>
     </policyType>
</policyTypeList>
```

Note: Any number of policy types can be registered in a single policy type registration file.

The following figure illustrates the policy type registration schema corresponding to the XML file given in an example above.



Running Policy Type Callbacks

Policy type callbacks are executables that are run at predetermined moments in the lifecycle of a policy of the specific type. There are four types of callbacks: Edit, Check, Deploy and Cleanup.

A number of variables can be used to define callbacks. Before the callback is executed, a variable replacement is performed. This allows runtime values to be passed to the callbacks as parameters.

For the detailed information about callbacks, see *HPOM Concepts Guide*. For more information about policy types, see the *HPOM Administration UI Online Help*.

Using ConfigFile Policies to Run Callbacks on Agents After Deployment

The ConfigFile policy type enables you to perform some post-processing (such as loading OV Composer factstores) by running callbacks on an agent after the ConfigFile policy deployment.

To enable running callbacks on an agent, perform the following:

1. On the management server, edit the ConfigFile policy so that it contains the following lines:

   ```
   SyntaxVersion=<SyntaxVersion>
   Application=<Application>
   SubGroup=<SubGroup>
   Filename=<Filename>
   CallFunctionClassID=
   Size=<Size>
   Data:
   #$Installcommand=<Installcommand>
   #$Commandtype=<Commandtype>
   # @(#)alarmdef             <product version> <date> for <OS> =*=
   #
   #  <alarm definitions>
   ...
   ```

   Where the values could be as in the following example:

   ```
   SyntaxVersion=1
   Application=Performance_agent
   SubGroup=Alarmdef_for_HP-UX
   Filename=hpux.alarmdef
   CallFunctionClassID=
   Size=14454
   Data:
   #$Installcommand=$OvPerlADir$/perl \
   $OvInstrumentationDir$/PostDeployActions.pl alarmdef \
   hpux.alarmdef
   #$Commandtype=3
   # @(#)alarmdef             05.00.000      01JUN2009 for PA/HP-UX
   =*=
   #
   #  HP Performance Agent alarm definitions
   ...
   ```

2. Assign and distribute the policy to a managed node.

   When the policy is deployed to the managed node, the policy body data stated below the line "`#$Commandtype=<Commandtype>`" is placed into a file, located at:

   `<OvDataDir>``/conf/`*`<Application>`*`/`*`<SubGroup>`*`/`*`<Filename>`*

   Where *`<Application>`*, *`<SubGroup>`* and *`<Filename>`* are defined in the policy body.

   In this example, the policy creates `/var/opt/OV/conf/Performance_agent/Alarmdef_for_HP-UX/hpux.alarmdef`.

For more information about the ConfigFile policy, its syntax and keywords, see the *HPOM Administration UI Help*. For more information on editing policies, see "Policy Configuration" on page 78. For details about assigning and distributing policies, see "Assigning Policies" on page 76 and "Deploying Policies" on page 76.

Assigning Policies

Policies can be assigned to policy groups using the `opcpolicy` command and to managed nodes using the `opcnode` command, as follows:

- Policy assignment to a policy group:

  ```
  # opcpolicy -add_to_group group=<policy_group> pol_name=<policy_name> pol_
  type=<policy_type_name> version=<policy_version>
  ```

- *Policy assignment to a node:*

  ```
  # opcnode -assign_pol pol_name=<policy_name> pol_type=<policy_type_name>
  version=<policy_version> node_name=<node_name> net_type=<node_network_type>
  ```

You can list policy types by using the `opcpoltype -list` command. To remove policy assignments, use the following options:

- `opcpolicy -del_from_group`
- `opcnode -deassign_pol`

To list the contents of a policy group, use the `opcpolicy` command as follows:

```
# opcpolicy -list_group pol_group=<policy_group_name>
```

To retrieve a list of policies assigned to a managed node, use the `opcnode` command as follows:

```
# opcnode -list_ass_pol node_name=<node_name> net_type=<node_network_type>
```

Example of assigning a policy to a policy group:

```
opcpolicy -add_to_group pol_name="Test policy" pol_type=Logfile_Entry version=1.0
pol_group="Test group"
```

Example of assigning a policy to a managed node:

```
opcnode -assign_pol pol_name="Measurement policy" pol_type=Measurement_Threshold
version=1.2 node_name=remote.hp.com net_type=NETWORK_IP
```

For more information about the parameters and options available with the `opcpolicy` and `opcnode` commands, see the *opcpolicy(1m)* and *opcnode(1m)* manual pages.

Deploying Policies

You can start the policy-distribution process as follows:

```
# /opt/OV/bin/OpC/opcragt -distrib -policies <node_name> [ <node_name> ... ]
```

By using the `-policies` option of the `opcragt` command you retrieve the assigned policies from the HPOM repository, prepare them for the distribution and start their deployment to the managed nodes.

For more information about the `opcragt` command options, see the *opcragt(1m)* manual page.

Deleting Policies

A single policy, as well as an entire container, can be removed from the database by using the `opcpolicy` command line tool. To delete a policy, it has to be uniquely identified by either its name/type/version combination or by its UUID. If just policy name and type are provided, the entire policy container is deleted.

Note: Deleting the policy also results in deleting all its assignments.

Following is an example of deleting a policy container:

```
# opcpolicy -remove pol_name="Test policy" pol_type=Logfile_Entry
```

Policy groups are deleted by using `-del_group` option of the `opcpolicy` command line utility. For example, to delete the policy group ”Test group”, use the following command:

```
# opcpolicy -del_group pol_group="Test group"
```

For more information about command options, see the *opcpolicy(1m)* manual page.

Downloading Policies

You can download policies using the `-download` option of the `opcpolicy` command line utility.

Note: If a policy version is omitted from the command line arguments, the entire container is downloaded.

Example of policy download using the `opcpolicy` command line tool:

```
# opcpolicy -download pol_name="Oracle messages" pol_type="Open_Message_Interface"
version=1.0 dir=/tmp
```

For more information about the parameters available with the `opcpolicy` command, see the *opcpolicy (1m)* manual page.

Downloading Policies with Categories

When downloading a policy by using the Administration UI, the related categories (that is, instrumentation) are not included in the download archive.

To download the related categories, add `WITH_POLICIES` to the `.dsf` file before running the `opccfgdwn` tool for downloading the configuration on the HP Operations management server. For example:

```
$ cat /tmp/test.dsf

POLICY_GROUP """"mypolgrp"""" WITH_POLICIES;
```

```
$ opccfgdwn /tmp/test.dsf /tmp/test
```

Policy Configuration

This section describes how you can use the tools provided to edit and modify policies to suit the needs of your particular environment. The section includes information and tips covering the following areas:

Registering a Policy Editor

The default policy types delivered with HPOM 9.xx have a defined editor. You can define a different editor for custom policy types when registering a new policy type.

To define and register an editor for a policy type, use the `opcpoltype` command with the following options:

```
# opcpoltype -reg -editor
```

For more information about registering new policy types, see "Registering Policy Types" on page 73.

Changing the Defined Policy Editor

To change the defined editor use the `opcpoltype` command line tool as shown in the following example:

```
# opcpoltype –editor –type “X policy type” /usr/local/bin/xeditor
```

For more information about command options, see the *opcpoltype(1m)* manual page.

Note: If the editor is running on a system other than the HPOM management server, make sure that the policy body is transferred back to the HPOM server, and that the upload is properly performed. To upload a new policy, after the editing is done, you can use either `opcpolicy -upload` or the provided APIs. For more information about the `opcpolicy` command options, see the *opcpolicy(1m)* manual page.

Editing Policies from the Command Line

You can edit policy bodies from the command line by using the `opcpoledit` command line utility. The `opcpoledit` command downloads the requested policy body. It can also run the registered editor and

uploads the policy back into the HPOM repository.

The syntax of `opcpoledit` is as follows:

```
opcpoledit { policy=<name> type=<type> version=<version> |
             id=<uuid> }
           [ part=<body number> | suffix=<body suffix> ]
           [ dir=<directory> ] [ verbose=yes ] [ run=yes ]
           [ upload=yes [ replace=yes] ]
```

By using the `verbose=yes` option you can open the command line interface for editing the policy body and instructions for upload after editing. By using the `run=yes` option you can run the registered editor.

If you use `upload=yes` together with `run=yes`, the policy is automatically uploaded after it is successfully edited. If you use `replace=yes`, the policy is uploaded to replace the existing policy in the database, otherwise a new minor version of the policy is created.

For more information, see the *opcpoledit(1m)* manual page.

Changing Policy Attributes

To change policy attributes such as description or policy body syntax, use the `opcpolicy` command line tool with `-update` option. `opcpolicy` can only be used to modify attributes that are part of the policy header and will not affect the contents of the policy bodies. For a list of the attributes that can be changed, see the *opcpolicy(1m)* manual page.

Changing the Policy Syntax Versions

Each policy type can have a different syntax version. If the syntax version is not specified in the command line arguments when registering a new policy, it is set to the default, 1.

You can change the syntax version of a policy by using the `opcpolicy -update` option with `syn=<syn>` argument set, for example:

```
# opcpolicy -update syn=<policy_syntax_version>
```

Note: After editing a policy, its syntax version is not changed by default. It can be automatically changed with the `opcpolicy -syn` command within check callback. If you want to change the syntax version manually without using the check callback, you must do it before the next deployment. Otherwise, the policy will be deployed with an old syntax version.

See the *opcpolicy(1m)* manual page for more information about the `opcpolicy` command options.

Policy Versions

With HPOM 9.xx, it is possible to have multiple versions of policies stored on the management server. Having multiple versions of policies on the HPOM 9.xx management server enhances the flexibility in

operating with policies and policy groups, and allows simplified interaction between HPOM for UNIX on Linux and Windows platforms.

- "Policy-Version Conflicts" on page 80
- "Changing the Policy Version" on page 80

For more information about the concepts of policy versions and the organization of the policy-group hierarchy, see the *HPOM Concepts Guide*.

To learn how to handle multiple versions of HPOM configuration (which includes, in addition to the policies, policy groups and the instrumentation data) on managed nodes, see "Managing Multiple Versions of HPOM Configuration on Managed Nodes" on page 147.

Policy-Version Conflicts

If one version of a policy is assigned directly to a managed node at the same time as another version of the same policy is assigned indirectly (for example, by assignment to a node group or policy group), it can sometimes be unclear which version of the assigned policy should actually be deployed to the managed node. HPOM assumes a higher priority for direct assignments than for any assignment made indirectly, for example, through groups. In other words, in the event of a conflict between directly and indirectly assigned policy versions, direct assignment to a managed node is considered more important (and overwrites) any indirect assignment even if the version specified by the indirect assignment is more recent.

For example, if policy version 1.3 is assigned directly to managed node AA and version 1.6 of the same policy is assigned to a node group to which managed node AA belongs, then any policy deployment would deploy version 1.3 of the policy in preference to version 1.6, which although more recent is assigned indirectly through a node group.

Caution: HPOM is not able to prioritize different versions of a policy if both versions are assigned indirectly (for example, to two different node groups or policy groups). To prevent unexpected deployment results, try to avoid assigning different versions of the same policy to different node groups or policy groups.

For more information about automating policy assignments and managing version conflicts, see "Policy-Assignment Updates" on page 81 and "Policy-Assignment Conflicts" on page 82.

Changing the Policy Version

All policies have version numbers. To replace a particular policy version, use the `opcpolicy` command with the `-update` parameter as follows:

```
# opcpolicy -update version=<policy_version>
```

You can also upload and download specific versions of a policy and instruct HPOM what to do if a version already exists. For more information about the `opcpolicy` command and the `-update` parameter, see the *opcpolicy(1m)* manual page.

The policy version numbers can also be changed without the need to modify the policy content. This is especially useful when aligning the policy versions that are released together. See the *opcpolicy(1m)* manual page for more information about command options.

Note: The new version number creation results in the creation of a new policy, even if the content is unchanged. The new policy has a new version UUID, but the container ID is same as before. On the other hand, changing the policy name results in a new object in the database with a new version UUID and a new container ID.

Policy Assignment Tasks in HPOM

Table 5 lists policy-related tasks and operations provided with HPOM. For more information about updating policy assignments, see *HPOM Concepts Guide*.

Table 5: Policy Management in HPOM

		HPOM Policies	
		New	**Existing**
Create		√	-
Deploy		√	√
Assign		√	√
Update assignments		-	√
Modify	Create new version	-	√
	Overwrite	-	√
	Force Overwrite	-	√

Policy-Assignment Updates

Policies can be assigned to nodes, node groups, and policy groups. The modification of the policy leads to a new policy version. This means that the existing assignments point to the older policy version, and not to the modified one. For this reason, the assignments must be updated.

You can configure HPOM to perform the policy-assignment updates automatically. There are three assignment modes, namely `FIX`, `LATEST`, and `MINOR_TO_LATEST`. The following list explains what the different modes mean:

`FIX`	Deploy a specific (fixed) version of the policy assigned to a node group, for example, version 1.3. The deployed policy version will not change even if newer versions of the policy become available.
`LATEST`	Deploy the most recent version of the assigned policy that is available, regardless of major (1.x) or minor (x.10) versions available. For example, if the original policy assignment was version 1.3 and version 2.5 is available at the time of deployment, then deploy policy version 2.5.
`MINOR_TO_LATEST`	Deploy the policy with the highest minor version in and the same major version as the originally assigned policy. Assigning a policy 1.3 with option `MINOR_TO_LATEST` will deploy the highest 1.x version at the time of deployment (for example, 1.5) but ignore any newer major versions (for example, 2.x or 3.x). If no policies are available with a minor version higher than 1.3, then version 1.3 will be deployed.

For more information about specifying the policy-assignment mode with the `opcpolicy` and `opcnode` command line utilities, see the *opcpolicy(1m)* and *opcnode(1m)* manual pages.

Potential Pitfalls with Assignment Modes

These are potential pitfalls with the policy assignment modes:

- With the `FIX` mode

  Even though a policy version is created and the configuration is deployed to a managed node, nothing changes because the assignment still points to the older policy version.

- With the `LATEST` mode

  Higher test versions could be deployed together with the automatic updates of the configuration to the production environments.

- With the `MINOR_TO_LATEST` mode

  With this mode, a rollback to an earlier version can lead to pitfalls. For more information, see "Rollback to Previous Versions" on page 156.

Policy-Assignment Conflicts

Conflicts can occur if there are differences between the version of a policy assigned directly to a managed node and other versions of the same policy that are assigned indirectly, for example, by

assignment to one or more node or policy groups to which the original managed node belongs. In the event of a version conflict, HPOM assumes a higher priority for policies that are directly assigned.

Note that HPOM is not able to prioritize different versions of a policy if both versions are assigned *indirectly* (for example, to two different node groups or policy groups). To prevent unexpected deployment results, try to avoid assigning different versions of the same policy to different node or policy groups.

For more information about conflicts between policy versions in the context of policy assignment, see "Policy-Version Conflicts" on page 80.

Database Reports

HPOM provides preconfigured reports for the administrator and the operators. In addition, you can create customized reports using the report writer supplied with the installed database or any other report-writing tool.

You can do the following with database reports:

- Display the generated report in a window.
- Save the generated report to a file.
- Print the generated report.

For instructions that describe how to add your own reports and reference information about the database schema which you will need if you want to create your own SQL reports, see the *HPOM Reporting and Database Schema*.

Generating Web-based Reports

You can retrieve specific information directly from the database and publish and view the resulting information in graphically rich formats, which are suitable for the Web-based reports. To generate these Web-based reports, use the enhanced reporting features of HPOM in conjunction with HP Service Reporter. For more information, see the documentation supplied with the HP Service Reporter and the *HPOM Concepts Guide*. You can also use HP Performance Insight to generate reports. For more information about generating and viewing reports with HP Performance Insight, see the product documentation. For more information about restrictions and supported configurations for Web-based reporting, see "Report Security" on page 89

Integrating a New Report

HPOM enables you to integrate self-written reports into the list of reports available for use to administrators and other users. You can integrate the reports in the following ways:

- Modify the `admin.rpts` file. For more information about the contents of the `admin.rpts` file, see "Defining Customized Administrator Reports" on page 88.
- Use the tools provided in the Administration UI.

You can create SQL reports (SQL scripts called from the shell script `call_sqlplus.sh`) or program reports. To modify a report you can either change the program, or customize the report's configuration file.

You configure a new report by creating a new script, writing a new program, or by creating a new SQL file. Then you edit existing plain text files to integrate the new reports. These configuration files define which reports are intended for the administrator and which are for the operator.

1. Change to the directory containing the report files. Enter:

   ```
   # /etc/opt/OV/share/conf/OpC/mgmt_sv/reports/<lang>
   ```

   Note: Alternatively, you can add your script to any location inside `/etc/opt/OV/share/conf/OpC/mgmt_sv/reports`. If the `call_sqlplus.sh` script does not find the specified report inside the current language directory, it searches for it in the `reports` directory.

2. Either modify an existing report, or create a new SQL report.

   The report file name must have the `.sql` (Oracle) or `.psql` (PostgreSQL) extension and must reside in the `reports` directory you accessed in the previous step.

3. Test the new or modified report.

   Use the `call_sqlplus` command with the following parameters:

   ```
   # /opt/OV/bin/OpC/call_sqlplus.sh <name> <parameter>
   ```

   In this instance, <*name*> is the name of the report file without the `.sql` suffix and <*parameter*> is an optional parameter passed to the report.

   For more information about report syntax, permitted parameters, and file-naming conventions, see "Report Syntax " on page 85 and "Report Parameters" on page 85.

4. Edit the `admin.rpts` report definition file.

   The `admin.rpts` configuration file lists reports for the HPOM administrator. It also contains a definition for each report. For more information about report syntax, permitted parameters, and file-naming conventions, see "Report Syntax " on page 85 and "Report Parameters" on page 85.

5. Save the appended `admin.rpts` report definition file.

   Save the new or modified report definition in the `reports` directory:`/etc/opt/OV/share/conf/OpC/mgmt_sv/reports/<lang>`.

Report Syntax

The syntax of the report definition is as follows:

DESCRIPTION	*%<descriptive text>*
PARM	*%<OpC parameter>*
REPORTFILE	*%<full directory path to file or program>*
REPORTNAME	*%<name>*
REPORTTYPE	*%<PGM>*

Report Parameters

When configuring reports generated by HPOM's reporting mechanisms `call_sqlplus.sh` or `opcmsgsrpt`, you can restrict the scope of the resulting report by using the parameters displayed in the following list:

`$node`	Selected node name (or ID)
`$nodegrp`	Selected node-group name (or ID)
`$msggrp`	Selected message-group name (or ID)
`$operator`	Selected operator name (or ID)
`$application`	Selected application ID
`$message_history`	Selected message ID
`$message_active`	Selected message ID
`$template`	Selected template

Note that the `call_sqlplus.sh` script expects the name of an object (rather than the ID). If you want to specify a name (rather than an ID) with the `opcmsgsrpt` command, you must use the `-n(ame)` option, as illustrated in the examples displayed in "Generating Reports with Command-Line Tools" on page 85.

Generating Reports with Command-Line Tools

To use the command-line tool `call_sqlplus.sh` to generate and display a detailed report listing all node assignments to a particular node group (for example, called "linux"), perform the following step:

```
# /opt/OV/bin/OpC/call_sqlplus.sh sel_ngrps linux
```

For more information about the SQL scripts run by the `call_sqlplus` script (for example, to specify all or selected applications, all or selected message groups, and so on), see Chapter 2.

To use the command-line tool `opcmsgsrpt` to generate and display a *detailed* report listing all *active* messages belonging to a particular user (for example, the HPOM user `opc_op`), perform the following step:

```
# /opt/OV/bin/OpC/opcmsgsrpt -n opc_op Active Detailed
```

Preconfigured Administrator Report Types

HPOM provides a range of preconfigured reports that list and describe all aspects of the current HPOM configuration and deployment, for example: nodes and node groups, applications and application groups, users and user profiles, and so on.

HPOM uses the `call_sqlplus.sh` script to access administrator reports. The `call_sqlplus.sh` script is located in `/opt/OV/bin/OpC/` along with other internal utilities that HPOM uses to run reports, for example, `opcmsgsrpt`. As the name suggests, the `call_sqlplus.sh` script runs an additional SQL script with any required parameters. For more information about the parameters available for use with the internal report-generating utility `opcmsgsrpt`, see "Report Parameters" on page 85.

The SQL scripts called by the `call_sqlplus.sh` script (for example, `msg_mgrp`, `cert_state`, and so on) are located in the directory `/etc/opt/OV/share/conf/OpC/mgmt_sv/reports/C`. The following table lists and briefly describes the reports configured for the HPOM administrator and indicates the command called to generate the report and any required parameters.

Preconfigured Reports for the Administrator

Report Name	Description	Command
All Active Messages	Report on the number of active messages per message group.	`call_sqlplus.sh msg_mgrp`
Cert. State Overview	Report about the status of security certificates assigned to all configured managed nodes.	`call_sqlplus.sh cert_state`
License Overview	A report about the availability and of HPOM licenses.	`ovolicense -r -p HPOM`
Node Config Report	Report about the assignment of all policies to managed nodes.	`call_sqlplus.sh node_conf`
Nodes Overview	Report about all configured nodes showing: the node name, the machine type, the node type (for	`call_sqlplus.sh all_nodes`

Preconfigured Reports for the Administrator , continued

Report Name	Description	Command
	example, message-allowed, controlled), the license, and any heartbeat-polling settings.	
Node Report	Detailed report about a selected managed node.	`call_sqlplus.sh sel_nodes $node`
Node Reference Report	Report about referenced nodes that are not in the node bank.	`call_sqlplus.sh node_ref`
Nodesgroup Overview	General report about *all* configured node groups.	`call_sqlplus.sh all_ngrps`
Nodegroup Report	Detailed report about an individual *selected* node group.	`call_sqlplus.sh sel_ngrps $nodegrp`
OMU Error Report	Report reviewing the HPOM error-log file (`System.*`) on the management server, in the following formats: Plain text: `/var/opt/OV/log/System.txt` Binary: `/var/opt/OV/log/System.bin`	`/bin/cat/var/opt/OV/log/System.txt`
Oper. Active Details	Report on all active messages for an operator (detailed description).	`opcmsgsrpt $operator, ACTIVE, DETAILED`
Oper. Active Message	Report on all active messages for an operator (short description).	`opcmsgsrpt $operator, ACTIVE`
Oper. History Messages	Short history of the (acknowledged) messages for a given operator.	`opcmsgsrpt $operator, HISTORY`
Oper. Pending Messages	Short description of pending messages for a given operator.	`opcmsgsrpt $operator, PENDING`
Operator Overview	Short description of all configured operators, including real and logon	`call_sqlplus.sh all_oper`

Preconfigured Reports for the Administrator , continued

Report Name	Description	Command
	names, role, rights, and responsibilities.	
Operator Report	Detailed report on a selected operator. Includes a responsibility matrix (node and message groups), available applications, and assigned user profiles.	`call_sqlplus.sh sel_oper $operator`
Templates Overview	List of *all* policies showing which policy group(s) the various policies belong to.	`call_sqlplus.sh all_templ`
User Profile Overview	Report on *all* configured user profiles.	`call_sqlplus.sh all_profiles`
User Profile Report	Detailed report on one *selected* user profile.	`call_sqlplus.sh sel_profile`

Defining Customized Administrator Reports

You can define or modify the list of administrator reports that are available by editing the `admin.rpts` file, which you can find in the following location:

`/etc/opt/OV/share/conf/OpC/mgmt_sv/reports/<lang>/admin.rpts`

If you try to save the output of HPOM administrator reports to a file but do not specify an absolute path (starting with a forward slash "/"), the file is saved by default in the directory of the UNIX user that started the HPOM administrator session. This directory is defined by $HOME or `/tmp` in that order. All files that you create and save as HPOM administrator are owned by the UNIX user who started the administrator's session; the user can (but does not have to) be the `root` user.

Generating Statistical and Trend-Analysis Reports

HPOM enables you to generate statistical and trend-analysis reports over a defined period of time. These reports can be configured to cover periods from as little as a few days to as much as weeks or even months.

Report Security

To enhance report security, HPOM restricts database access, network access, and web reporting capabilities. You can customize these security measures to match the particular needs of your organization.

- Database Access:

  For report-writing tools, HPOM restricts database access to a single database user, `opc_report`. This user has read-only access. The `opc_report` user makes use of `opc_report_role`. This report role is a kind of database user profile. You can use the role to enable additional users to access the database so they can create reports using information in the HPOM database tables.

- Network Access:

  Both Oracle and PostgreSQL databases have methods to restrict which hosts are allowed to establish a connection to the database. You can customize the database options to tighten security even further. For more information, see the database product documentation.

- Web Reports:

  To restrict access to web reports, HPOM requires you to place the web-reporting server on the same side of your firewall as the HPOM database server. HPOM does not support any other configuration. For example, you cannot open up the database server port on the firewall in order to allow access to the database for reports generated with the HP Reporter.

Flexible Management Configuration

This section contains the information required for administering the flexible management policies, which is presented as follows:

- "Locations of Flexible Management Policies" on page 90
- "Types of Flexible Management Policies" on page 90
- "HTTPS-based Event Forwarding" on page 91
- "Distributing Configuration and Policies in the Flexible Management Environment" on page 94
- "Integrating a New Management Server into an Existing Flexible Management Environment" on page 97

For more information about the HPOM flexible management environment, see the *HPOM Concepts Guide*.

For more information about the keywords, syntax, policy schedules, message forwarding policies, time policies and examples of flexible management policies, see the *HPOM Administration UI Help*.

Locations of Flexible Management Policies

HPOM provides a set of plain text policies you can use to configure and implement flexible management in a widely-distributed environment. These text policies are located in the following directory:

`/etc/opt/OV/share/conf/OpC/mgmt_sv/tmpl_respmgrs`

In this directory, you can also find a comprehensive `ReadMe` file that explains in great detail different types of policies.

Types of Flexible Management Policies

Table 6 provides a brief description of each policy.

Table 6: Example Policies for HPOM Flexible Management

Policy Name	Description
`backup-server`	Defines the responsible managers for an HPOM backup server. If the HPOM primary server fails, management responsibility can be switched to a backup server. The policy defines two management servers: `M1` and `M2`. Management server `M2` can act as a backup server for management server `M1`. Note that HPOM also supports the configuration and use of server pooling, which also provides backup functionality.
`example.m2`	Combines follow-the-sun and service-oriented message distribution functions.
`example.m3`	Additional example policy for follow-the-sun functions.
`followthesun`	Defines the time policies and responsible managers for HPOM follow-the-sun responsibility switching. The policy defines three management servers: `M1`, `M2`, and `M3`. These management servers can switch responsibility at different times of the day and week.
`hier.specmgr`	Provides an example of hierarchical management responsibility. SNMP traps are sent to the local management server. All other messages are sent to the primary management server.
`hier.time.all`	Provides an example of hierarchical management responsibility. Responsibility is switched between two servers according to a follow-the-sun time policy.

Example Policies for HPOM Flexible Management, continued

Policy Name	Description
hier.time.spec	Provides an example of hierarchical management responsibility. SNMP traps are sent to the local management server. All other messages are sent to the primary management server according to a follow-the-sun time policy.
hierarchy.agt	Defines the responsible managers for hierarchical management responsibility switching for all nodes. The policy defines two management servers: M1 and MC. M1 is configured as the primary manager for all nodes. MC is configured as an action-allowed manager for all nodes.
hierarchy.sv	Defines the responsible managers for hierarchical management responsibility switching for regional management servers.
msgforw	Defines the responsible managers for manager-to-manager message forwarding. The policy defines the message forwarding target rules.
outage	Defines the period of time in which a service is to be provided, or in which a system (for example, a database server) or a service is scheduled to be unavailable.
service	Defines the responsible managers for service-related message distribution (for example, competence centers where experts in particular technical areas are available). The policy defines a local management server: M1. The policy also defines two examples of service centers: a database service center (DBSVC) and an application service center (ASVC).

HTTPS-based Event Forwarding

HPOM uses HTTPS-based communication to forward events in a flexible management environment. HTTPS-based event forwarding establishes a high level of security for the communication between management servers in an HPOM environment.

Enabling HTTPS-based Forwarding

To enable HTTPS-based event forwarding, you must establish a trust relationship between the HP Operations management servers that will be communicating directly.

For more detailed information about how to set up a trust relationship between HP Operations management servers, see the HP Operations Agent documentation.

To disable HTTPS-based event forwarding, set the parameter to false.

For faster HTTPS-based event forwarding set the configuration setting `OPC_DONT_FORW_MSGKEY_ACK` to `TRUE`:

```
# ovconfchg -ovrg server -ns opc -set OPC_DONT_FORW_MSGKEY_ACK TRUE
```

In this case, the acknowledge and annotation add change events caused by message key relations are not forwarded. If the target server has the same messages, it already handles the same message key relation. If this flag is not set to `TRUE` (by default), the correlation is performed twice, which may lead to lock time-outs and duplicate annotations.

Configuring HTTPS-based Forwarding

Although the default values will be adequate for most needs, you can reconfigure HTTPS-based message forwarding to suit your needs.

The parameters listed in Table 7 let you configure different aspects of event forwarding. See "Message Forwarding Configuration Parameters" on page 92 for more information about each parameter.

Table 7: Event Forwarding Configuration Parameters

Parameter Name	Default value	Description
`MAX_DELIVERY_THREADS`	10	Maximum number of delivery threads
`MAX_INPUT_BUFFER_SIZE`	100000	Maximum size of the internal input buffer (bytes)
`MAX_FILE_BUFFER_SIZE`	0 (unlimited)	Maximum size of the buffer file on disk (bytes)
`BUFFER_PATH`	`/var/opt/OV/share/tmp/OpC/mgmt_sv/snf`	Directory for buffering files
`REQUEST_TIMEOUT`	3600	Time after which a request time-outs and will not be delivered to remote servers (seconds)

Message Forwarding Configuration Parameters

`MAX_DELIVERY_THREADS`	Determines the maximum number of delivery threads that the forward manager will create when using HTTPS-based message forwarding. It is recommended to leave this variable at its default value, unless your environment contains a large number of servers to which

	messages are forwarded and you experience performance problems with forwarding.
MAX_INPUT_BUFFER_SIZE	Determines the size of the memory buffer used by the forward manager (in bytes). There is no need to change this value, unless issues with the delivery of very large messages occur.
MAX_FILE_BUFFER_SIZE	Determines the maximum size of the buffer file on a disk, used by the forward manager to store messages that are to be delivered to remote HP Operations management servers that are currently inaccessible. Increase this value if you expect frequent communication failures between HP Operations management servers and usually transfer large amounts of messages.
BUFFER_PATH	Determines the location of the directory in which the forward manager stores buffer files. Change this location only if you experience loss of messages and need to place the buffer files on a file system with more disk space.
REQUEST_TIMEOUT	Time limit after which undeliverable messages and message operations are discarded. Increase this value if you expect frequent communication failures that last longer than one hour. Set this value to 0 for unlimited queuing of messages and message operations.

Changing Parameter Values

The parameters listed in Table 7 are located in the `opc.opcforwm` namespace. To change their values, use the `ovconfchg` command line tool.

For example, if you want to limit the size of the buffer file on the disk to 200000 bytes, use the following command:

```
# ovconfchg -ovrg server -ns opc.opcforwm -set MAX_FILE_BUFFER_SIZE 200000
```

To check the current values of the HTTPS-based forwarding parameters, use the following command:

```
# ovconfget -ovrg server opc.opcforwm
```

Note that only the non-default values are displayed.

Troubleshooting Message Forwarding Problems

If you need to remove all buffered messages, perform the following steps:

1. Stop the HP Operations management server processes:

   ```
   # ovc -stop OPC
   ```

2. Remove the directory in which the forwarding manager stores buffered files:

   ```
   # rm -rf /var/opt/OV/share/tmp/OpC/mgmt_sv/snf
   ```

3. Start the HP Operations management server processes:

   ```
   # ovc -start OPC
   ```

Distributing Configuration and Policies in the Flexible Management Environment

Distributing relevant configurations and policies to all relevant management servers and nodes simplifies centralized product development. You can develop configurations and policies on the central server, and then distribute them to designated servers and managed nodes.

When policies are deployed to the HTTPS agents, they are provided with an owner string. Primary and backup management servers should share the same owner string.

If you want to start using the backup management server, overwrite the default owner string by using the following command on the backup management server:

```
ovconfchg -ovrg server -ns opc -set OPC_POLICY_OWNER <primary_server_fully_qualified_name>
```

You can also change the `OPC_POLICY_OWNER` string to any desired value but the values must be identical on both management servers.

Note: Only one owner string can be set per management server.

If the backup management server is not set up in exactly the same way as the primary management server, the agent may be configured differently when policies and instrumentation files are deployed. Instrumentation files from the primary management server remain, if not overwritten by the backup management server. Additional instrumentation files from the backup management server will be deployed and cumulated on the agent. Policies are replaced only if the primary and the backup management server use the same owner string or if the policies are identical. All other policies on the agent will remain unchanged, because they belong to different owners.

mgrconf and nodeinfo Policies in Flexible Management Environments

The `mgrconf` and `nodeinfo` policies are treated as special cases. The rules described in the section "Dealing with Identical Policies Deployed by Different Management Servers" on page 95 are not applicable to these two policies.

There can be only one instance of either of these policies per managed node. The management server that deploys either of these policies first remains the owner forever. The second management server cannot overwrite the existing policy. Therefore, it is recommended that in the competence center scenarios, only one server is used to deploy the `mgrconf` policies.

You can change the `owner` attribute for `mgrconf` and `nodeinfo` on a managed node, by using the `opcdeploy` command on the management server or the `ovpolicy` command on the management node.

Changing the owner Attribute on the Management Server

To change the `owner` attribute in `nodeinfo`, run the following command:

```
opcdeploy -cmd "ovpolicy -setowner OVO:<fully_qualified_mgmt_server_name> -poltype configsettings" -node <managed_node_name>
```

To change the `owner` attribute in `mgrconf`, run the following command:

```
opcdeploy -cmd "ovpolicy -setowner OVO:<fully_qualified_mgmt_server_name> -poltype mgrconf" -node <managed_node_name>
```

Changing the owner Attribute on the Managed Node

To change the `owner` attribute in `nodeinfo`, run the following command:

```
ovpolicy -setowner OVO:<fully_qualified_mgmt_server_name> -poltype configsettings
```

To change the `owner` attribute in `mgrconf`, run the following command:

```
ovpolicy -setowner OVO:<fully_qualified_mgmt_server_name> -poltype mgrconf
```

Dealing with Identical Policies Deployed by Different Management Servers

Policies are identified using their IDs and policy name, type, and version. If an ID is present, it has a higher priority than a name with policy type and version.

Identical policies are determined in the following way:

- The same policy ID
- The same policy name, type, and version, but different policy ID.

Identical policies can be modified by multiple management servers, independent of the policy owner. Consequently, many instances of the same policy are not installed on one agent and multiple messages are not created for the same issue.

If you use multiple management servers to deploy the same configuration data, they are acting as backup management servers, and their data should be synchronized.

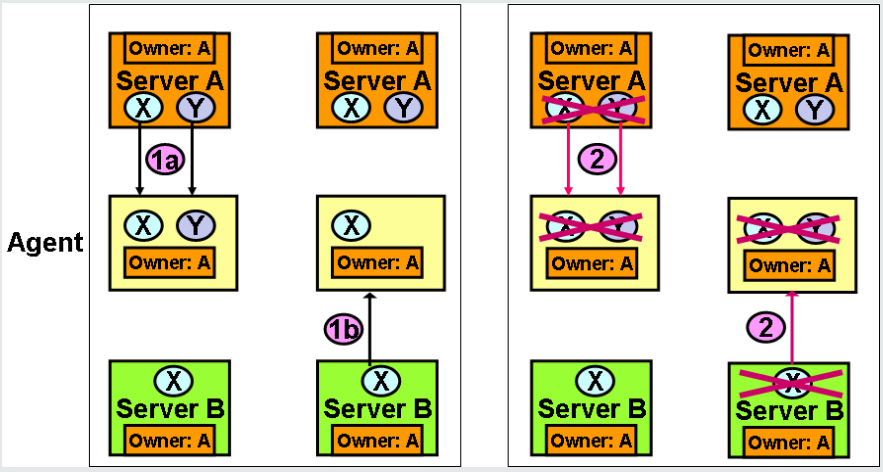
The `delta` and `force` distribution modes are available for the flexible management environments. `force` replaces all policies of the calling owner and all identical policies, even though they are owned by different management servers.

For the non-identical policies, in the delta and in the force modes, a de-assigned policy is only removed by the same owner.

The following examples show how the policies are handled between multiple configuration servers.

Server A and Server B use the same owner String “A”

Assume that there is a management server A, a management server B, a policy X, and a policy Y. Policy X is assigned to the agent from both management server A and management server B. Policy Y is assigned to the same agent only from management server A. Both management server A and management server B use owner string “A”.



1. Trigger configuration distribution.
   - **From server A, in the delta mode and in the force mode:** Policy X and policy Y are deployed and have owner “A”.
   - **From server B, in the delta mode:** Nothing is changed for policy X. It still has owner “A”. Policy Y is removed.

     From server B, in the force mode: Policy X is overwritten and still has owner “A”. Policy Y is removed.
2. Deassign policy X and trigger distribution.
   - **From server A, in the delta mode and in the force mode:** Policy X is removed.
   - **From server B, in the delta mode and in the force mode:** Policy X is removed.

3. Deassign policy Y from server A, in the delta mode and in the force mode: Policy Y is removed.

Removing Policies in the Backup Management Server Scenario

This example shows how the policies can be removed from the agent in the backup server scenario:

1. The primary management server A deploys policy PA to agent G. Thus policy PA has owner A.
2. The backup management server B deploys the same policy PA to the same agent G. Because the policies are identical, the previously installed policy PA with owner A is removed and reinstalled from the backup management server B. Consequently, the reinstalled policy PA has owner B.
3. On the primary management server A, deassign policy PA and issue policy distribution to the same agent G.

The result is that policy PA is *not* removed from agent G, the policy PA has owner B. Thus only the backup management server B can remove it.

Integrating a New Management Server into an Existing Flexible Management Environment

Synchronizing messages in a flexible management environment when a new management server is to be added can be done by using the `opcactdwn` and `opcactupl` command line interfaces for downloading and uploading active messages.

With the `-target_mgmtsv` option that is available with `opcactdwn`, active messages in the database are prepared to be handled as “forwarded” messages. This means that later message operations such as adding annotations, acknowledging, and owning are synchronized.

To integrate a new management server into an existing flexible management environment, follow these steps:

1. Install the new management server and upload configuration data.
2. Stop the HP Operations management server processes on the new management server.
3. Clear the active messages on the new management server by running the following command:

   ```
   /opt/OV/bin/OpC/opcdbinst -act
   ```

4. Prepare or update the `msgforw` file for the old and new management servers (for example, add the new management server to the old management server’s `msgforw` file).

5. Activate the message forwarding modification on the old management server or management servers by using `ovconfchg`. New incoming messages and message operations are buffered for the new management server on the old management server or management servers.
6. Copy the `msgforw` file to the new management server.
7. Download the active messages by running the following command on the old management servers:

   ```
   opcactdwn -file <act_msgs> -target_mgmtsv <new_server>
   ```

   By doing this, existing messages are prepared for later message operations such as adding annotations, acknowledging, and owning so that they can be synchronized from each old management server to the new management server.
8. Copy the active message download files from one of the old management servers (whichever you choose) to the new management server.
9. Upload the active messages on the new management server by running the following command:

   ```
   opcactupl <act_msgs_file>
   ```
10. Start the HP Operations processes on the new management server by running the following command:

    ```
    ovc -start
    ```

Note: The same procedure can be used later on to synchronize management servers if they are out of synchronization (for example, if queue files or the SnF forwarding buffer on the target management server had to be cleared).

For detailed information about the `opcactdwn` and `opcactupl` command line interfaces, see the *opcactdwn* and *opcactupl* manual pages.

HPOM Variables

This section lists and defines the variables that can be used with HPOM, and gives an output example, where appropriate. Each variable is shown with the required syntax.

Types of Variables Supported by HPOM

HPOM supports the following types of variables:

- Environment variables:

  Variables for the shell environment. These variables can be set before starting HPOM.
- Configuration variables:

Variables for configuring the HP Operations management server and HTTPS agents.

- Variables in all message-source policies:

  Variables must be enclosed with angle brackets. If the HPOM agents cannot resolve a variable, the variable itself is displayed in the GUI.

- Variables in instruction-text interface calls:

  Variables can be used when calling the instruction text interface in the Java GUI

- Variables in application calls and the user interface:

  Variables can be used when calling applications or issuing a broadcast command, or can be passed to external programs. Do not use angle brackets with these variables.

- Variables used with Service Navigator

Note: It is also often useful to enclose the variable in quotes (“”), especially if the variable might return a value that contains spaces.

HPOM and User-Defined Variables

HPOM and user-defined variables can be used to compose messages, or can be passed as parameters to action calls. They can also be passed to external applications, by using the instruction text interface. Note that HPOM variables are reserved words that cannot be used for any other purpose, for example, creating user-defined variables.

A variable is defined simply by assigning a matched string to it. Variables must be delimited by the use of angle brackets (<>).

The following example shows a user-defined variable, error_text followed by an HPOM variable `$MSG_APPL`, used for obtaining the name of the application associated with the message:

```
/tmp/example_command <error_text> <$MSG_APPL>
```

Environment Variables

You can use the following environmental variables before starting HPOM.

`$OPC_BRC_HISTSIZE`	Returns the value of the environment variable for the length of the user’s broadcast command history. The default number of commands saved is 128 per user. Example: `export OPC_BRC_HISTSIZE=512`

Configuration Variables

For a complete list of the HPOM server configuration variables, see the *HPOM Server Configuration Variables*.

HPOM provides an automatic synchronization of the most configuration variables after a change of the HPOM configuration. Most configuration variables used in server processes (`opcdispm`, `opcmsgm`, `ovoareqsdr`, `opcforwm`, `opcactm`, `opcttnsm`) are updated automatically each time the `ovconfchg` command is used.

Some configuration variables used in the server processes are exceptions and are not always synchronized. The configuration variables that represent file and path names, queues and pipes names, port ranges, and pid files are rather set at process startup and are not usually synchronized automatically. However, the variables for the file names of the following configuration files are synchronized automatically:

- Outage policy:

  `OPC_OUTAGE_TEMPLATE` (default: `outage`)
- Message forward policy:

  `OPC_MSG_FORW_TEMPLATE` (default: `opcforw`)
- MSI configuration file:

  `OPC_MSI_CONF` (default: `msiconf`)
- Remote action filter configuration file:

  `OPC_ACTSEC_FILTER` (default: `remactconf.xml`)

The following configuration variables are set only at startup and are never updated online:

- `OPC_OPCCTLM_START_OPCSVCAM`
- `_M_ARENA_OPTS`
- `_M_SBA_OPTS`

The following configuration variables have a specific behavior:

`OPC_RQS_NUM_AGT_WORKERS`	Updated online, only if the value is increased.
`OPC_BBCDIST_RETRY_INTERVAL`	Might not update till the end of the previous interval.

Variables in Message Source Policies

You can use the following variables in most text entry fields (except where noted) for log files, the HPOM message interface, the threshold monitor, and the SNMP-trap policy. You can use the variables within HPOM, or pass them to external programs. To ensure correct processing, you must enter the variables with the angle brackets. For details on policy body grammar, see the *HPOM Concepts Guide*.

`<$MSG_APPL>`	Returns the name of the application associated with the message. This variable cannot be used in log-file policies. The variable returns

	the default object, not the object set in the conditions window. Sample output: `/usr/bin/su(1) Switch User`
`<$MSG_GEN_NODE>`	Returns the IP address of the node from which the message originates. Sample output: `14.136.122.123` for IPv4, `fec0::94f6:cff:fe4d:ccdd` for IPv6
`<$MSG_GEN_NODE_NAME>`	Returns the name of the node on which from which the message originates. Sample output: `richie.c.com`
`<$MSG_GRP>`	Returns the default message group of the message. Sample output: `Security`
`<$MSG_ID>`	Returns the unique identity number of the message, as generated by the message agent. Suppressed messages do not have message IDs. Sample output: `6e998f80-a06b-71d0-012e-0f887a7c0000`
`<$MSG_NODE>`	Returns the IP address of the node on which the event took place. Sample output: `14.136.122.123` for IPv4, `fec0::94f6:cff:fe4d:ccdd` for IPv6
`<$MSG_NODE_ID>`	Returns the name of the node on which the event took place. Sample output: `richie.c.com` This variable is only available in the `Service Name` field.
`<$MSG_NODE_NAME>`	Returns the name of the node on which the event took place. This is the name returned by the node's name service. Sample output: `richie.c.com`

<$MSG_OBJECT>	Returns the name of the object associated with the event. This is set for the SNMP policy. This variable cannot be used in log-file policies. The variable returns the default object, not the object set in the conditions window.
<$MSG_SERVICE>	Returns the service name associated with the message. This variable can also be used for automatic and operator-initiated actions. Sample output: Application_Server
<$MSG_SEV>	Returns the default value for the severity of the message. This is set for the Logfile and OPCMSG policies. Sample output: Normal
<$MSG_TEXT>	Returns the original text of the message. This is the source text that is matched against the message text pattern in each condition. This variable returns an empty string when used in threshold monitor policies. Sample output: SU 03/19 16:13 + ttyp7 bill-root
<$MSG_TIME_CREATED>	Returns the time the message was created in seconds since January 1, 1970. Sample output: 950008585
<$MSG_TYPE>	Returns the default name set for Message Type. This name is set with the keyword MSGTYPE in the policy body.
<$OPTION(N)>	Returns the value of an optional variable that is set by opcmsg or opcmon (for example, <$OPTION(A)> <$OPTION(B)>, and so on). To find out how to set this variable, see the *opcmsg(1)* or *opcmon(1)* manual pages. The $OPTION variable cannot contain double quotes. Use single quotes instead.

Note: The <$NAME>, <$FULLNAME>, and <$SRCNAME> policy variables can only be used in measurement threshold policies.

Variable Resolution in HPOM

The variables used in HPOM can take one of several values, depending on the incoming message, default policy configuration or the configuration of the condition that the variables are matching. HPOM resolves the variable according to a specific order of priority.

Understanding Variable Resolution

HPOM calculates and sets the value of a variable according to the following order:

1. Use the value set by the external source (API/executable, event, and so on). For example, the `opcmsg` command with the following options to assign the value APP to the variable `<$MSG_APPL>`:

   ```
   # opcmsg app=APP object=O msg_text="Message text"
   ```

2. If the variable cannot be set by external sources, use a value generated by HPOM, for example, message ID.
3. If none of the above is valid for a variable, HPOM uses the value set in the policy body for which the variable is evaluated. If there is no default value set, set the value of the variable to 0 (zero) or leave it empty, depending on its type.

Note that HPOM adheres strictly to the specified order when resolving variable values. For example, if a value for `<$MSG_OBJECT>` is set by an external source (step 1), a default value set in step 3 is ignored.

Variables for Actions Only

The following variables can only be used in the `Node` field of *operator-initiated actions*, except for the variable `<$OPC_MGMTSV>` which can be used in all fields.

The variables `<$OPC_MGMTSV>`, `<$OPC_GUI_CLIENT>` and `<$OPC_GUI_CLIENT_WEB>` must be entered with angle brackets.

The variables must not be part of a string or be nested.

`$OPC_ENV(env variable)`	Returns the value of the environment variable for the user who has started HPOM. This variable is only available for operator-initiated actions. It is resolved in the action call. Sample output: `PATH, NLS_LANG, EDITOR, SHELL, HOME, TERM`. For example, if `SHELL` is set to `/usr/bin/ksh` and you have set up the operator-initiated action `echo $OPC_ENV(SHELL)`, the following command will be executed as operator initiated action: `echo /usr/bin/ksh`.

<$OPC_GUI_CLIENT>	Executes the application or action on the client where the Java GUI is currently running. This variable is resolved differently, depending on whether the GUI client is running on a UNIX-based system with DNS or on a PC using Microsoft Windows Internet Name Service (WINS). If you are using WINS, <$OPC_GUI_CLIENT> returns the WINS hostname.
<$OPC_MGMTSV>	Returns the name of the current HP Operations management server. This variable can be used in all fields related to actions. Sample output: richie.c.com
<$OPC_GUI_CLIENT_WEB>	Starts a web browser on the client where the Java GUI is currently running. This variable is resolved differently, depending on whether the GUI client is running on a UNIX-based system with DNS or on a PC using MS WINS (Windows Internet Name Service). If you are using WINS, <$OPC_GUI_CLIENT_WEB> returns the WINS hostname.
$OPC_USER	Returns the name of the HPOM user who is currently logged in on the management server. This variable is only available for operator-initiated actions. It is resolved in the action call. Sample output: opc_adm

Variables for Log-File-Encapsulator Policies Only

Note: These variables cannot be used with W2K8 and above.

You can use the following variables for most text entry fields in log-file policies. You can use the variables within HPOM, or pass them to external programs.

<$1>	Policies of Windows Event Log type. Returns one or more of the possible parameters that are part of a Windows event (for example, <$1> returns the first parameter, <$2> returns the second parameter, and so on.)
<$EVENT_ID>	Policies of Windows Event-Log type. Returns the event ID of the Windows event. <$EVENT_ID> simplifies the processing of multi-line

	event-log messages. You need the `Source` field and `<$EVENT_ID>` of the event to identify the event uniquely. Sample output: `0x0000600F`
`<$LOGFILE>`	Returns the name of the monitored log file. Sample output: `sulog`
`<$LOGPATH>`	Returns the full path to the monitored log file including the file name. Sample output: `/var/adm/sulog`

Variables for Threshold Monitor Policies Only

You can use the following variables in most text entry fields (exceptions are noted) of threshold monitor policies. You can use the variables within HPOM, or pass them to external programs.

`<$NAME>`	Returns the name of a threshold monitor. This name is set in the Monitor Name field of the Add/Modify Monitor window. You can use only alphabetical characters, numbers, underscore, and hyphen in the threshold monitor name. Other characters (for example, blank spaces or localized characters are not allowed). This variable cannot be used in the Monitor Program or MIB ID field. Sample output: `cpu_util`
`<$THRESHOLD>`	Returns the value set for a monitor threshold. This value is set in the `Threshold:` field in the `Condition No.` window. Sample output: `95.00`
`<$VALAVG>`	Returns the average value of all messages reported by the threshold monitor. Sample output: `100.00`
`<$VALCNT>`	Returns the number of times that the threshold monitor has delivered

	a message to the browser. Sample output: `1`
`<$VALUE>`	Returns the value measured by a threshold monitor. Sample output: `100.00`

Variables for SNMP Trap Policies Only

You can use the following variables in most entry fields (exceptions are noted) for SNMP trap text. You can use the variables within HPOM, or pass them to external programs.

`<$#>`	Returns the number of variables in an enterprise-specific SNMP trap (generic trap 6 Enterprise specific ID). Sample output: `2`
`<$*>`	Returns all variables assigned to the trap. Sample output: `[1] .1.1 (OctetString): arg1 [2] .1.2 (OctetString): kernighan.c.com`
`<$@>`	Returns the time the event was received as the number of seconds since the Epoch (January 1, 1970) using the *time_t* representation. Sample output: `859479898`
`<$1>`	Returns one or more of the possible trap parameters that are part of an SNMP trap (for example, `<$1>` returns the first variable, `<$2>` returns the second variable, and so on).
`<$\>1>`	Returns all attributes greater than *n* as *value* strings, which are useful for printing a variable number of arguments. `<$\>0>` is equivalent to `$*` without sequence numbers, names, or types. Sample output: `richie.c.com`
`<$\>+1>`	Returns all attributes greater than *n* as *name:value* string.

	Sample output: `.1.2: richie.c.com`
`<$+2>`	Returns the nth variable binding as *`name:value`*. This variable is not valid in the command field. Sample output: `.1.2: richie.c.com`
`<$\>-n>`	Returns all attributes greater than *n* as *`[seq] name (type): value`* strings. Sample output: `[2] .1.2 (OctetString): kernighan.c.com`
`<$-2>`	Returns the nth variable binding as *`[seq] name-type:value`*. This variable is not valid in command field. Sample output: `[2] .1.2 (OctetString): richie.c.com`
`<$A>`	Returns the node which produced the trap. Sample output: `richie.c.com`
`<$C>`	Returns the community of the trap. Sample output: `public`
`<$E>`	Returns the enterprise ID of the trap. Sample output: `private.enterprises.hp.nm.openView.hpOpenView`
`<$e>`	Returns the enterprise object ID. Sample output: `.1.3.6.1.4.1.11.2.17.1`
`<$F>`	Returns the textual name of the remote machine where the pmd is running, if the event was forwarded. Sample output: `kernighan.c.com`

<$G>	Returns the generic trap ID. Sample output: 6
<$N>	Returns the event name (textual alias) of the event format specification used to format the event, as defined in the Event Configurator. Sample output: OV_Node_Down
<$O>	Returns the name (object identifier) of the event. Sample output: private.enterprises.hp.nm.openView.hpOpenView.0.58916872
<$o>	Returns the numeric object identifier of the event. Sample output: .1.3.6.1.4.1.11.2.17.1
<$R>	Returns the true source of the event. This value is inferred through the transport mechanism that delivered the event. Sample output: kernighan.c.com
<$r>	Returns the implied source of the event. This may not be the true source of the event if the true source is a proxy for another source, such as when a monitoring application running locally is reporting information about a remote node. Sample output: richie.c.com
<$S>	Returns the specific trap ID. Sample output: 5891686
<$s>	Returns the event's severity. Sample output: Normal

<$T>	Returns the trap time stamp. Sample output: 0
<$V>	Returns the event type, based on the transport from which the event was received. Currently supported event types are SNMPv1, SNMPv2, SNMPv2C, CMIP, GENERIC, and SNMPv2INFORM. Sample output: SNMPv1
<$X>	Returns the time the event was received using the local time representation. Sample output: 17:24:58
<$x>	Returns the date the event was received using the local date representation. Sample output: 03/27/97

Variables in Scheduled-Action Messages

You can use the following variables in the `Scheduled Action - Start/Success/Failure Message` windows of scheduled action policies. You can use the variables within HPOM, or pass them to external programs.

<$PROG>	Returns the name of the program executed by the scheduled action policy. Sample output: opcsv
<$USER>	Returns the name of the user under which the scheduled action was executed. Sample output: root

Variables for Instruction-Text Interface Calls

The following variables can only be used in instruction text interface calls executed on the Java operator GUI.

<LOCAL_ON_JAVA_CLIENT>	Starts a program or script on the client where the Java GUI is currently running as a result of the instruction text interface call. For example, to start Microsoft Internet Explorer on the Java GUI client, use the following with the INSTR_INTERF_CALL argument in the file used as input to the opcinstr command line tool: <LOCAL_ON_JAVA_CLIENT> "C:\Program Files\ Internet Explorer\IEXPLORE.EXE"
<LOCAL_ON_JAVA_CLIENT_WEB>	Starts a web browser on the client where the Java GUI is currently running as a result of the instruction text interface call. For example, to start a web browser on the Java GUI client at the URL http://www.hp.com, use the following with the INSTR_INTERF_CALL argument in the file used as input to the opcinstr command line tool: <LOCAL_ON_JAVA_CLIENT_WEB> http://www.hp.com Depending on the configuration of the Java GUI work space, either the embedded or an external web browser is started.

For information about the command-line interface to the instruction-text interface, see the *opcinstrif (1m)* manual page.

Variables in Application Calls and the User Interface

You can use the following variables listed in most application text entry fields (exceptions are noted) of the GUI. You can use the variables within HPOM, or pass them to external programs.

$OPC_ENV(env variable)	Returns the value of the environment variable for the user who has started HPOM. Sample output: PATH, NLS_LANG, EDITOR, SHELL, HOME, TERM.
$OPC_EXT_NODES	Returns the node pattern of all external nodes that are selected at the time the application is executed. The names are separated by

	spaces.
$OPC_MSG_NODES	Returns the names of all nodes on which the events that generated currently selected messages took place. The names are separated by spaces. The nodes do not need to be in the node bank. If the same message is selected in more than one of these browsers, the duplicate selections is ignored. In the HPOM Java GUI, only nodes of the messages currently selected in the topmost browser are returned. Sample output: `kernighan.c.com richie.c.com`
$OPC_MSG_GEN_NODES	Returns the names of all nodes from which currently selected messages were sent by HPOM agents. The names are separated by spaces. The nodes do not need to be in the node bank. If the same message is selected in more than one of these browsers, the duplicate selections are ignored. In the HPOM Java GUI, only nodes of the messages currently selected in the topmost browser are returned. Sample output: `kernighan.c.com richie.c.com`
$OPC_MSG_IDS	Returns the Message IDs (UUIDs) of the messages currently selected in one or more open Message Browsers. If the same message is selected in more than one browser, the duplicate selections are ignored. In the HPOM Java GUI, only Message IDs of the messages currently selected in the topmost browser are returned. Sample output: `85432efa-ab4a-71d0-14d4-0f887a7c0000` `a9c730b8-ab4b-71d0-1148-0f887a7c0000`
$OPC_MSGIDS_ACT	Returns the Message IDs (UUIDs) of the messages currently selected in the Active/All and any HP Software Message Browsers. If the same message is selected in more than one of these browsers, the duplicate selections are ignored. In the HPOM Java GUI, only Message IDs of the messages currently selected in the topmost browser are returned. Sample output: `85432efa-ab4a-71d0-14d4-0f887a7c0000` `a9c730b8-ab4b-71d0-1148-0f887a7c0000`

$OPC_MSGIDS_HIST	Returns the Message IDs (UUID) of the messages currently selected in the History Message Browser. In the HPOM Java GUI, only Message IDs of the messages currently selected in the topmost browser are returned. Sample output: `edd93828-a6aa-71d0-0360-0f887a7c0000` `ee72729a-a6aa-71d0-0360-0f887a7c0000`
$OPC_MSGIDS_PEND	Returns the Message IDs (UUID) of the messages currently selected in the Pending Messages Browser. In the HPOM Java GUI, only Message IDs of the messages currently selected in the topmost browser are returned. Sample output: `edd95828-ac2a-71d0-0360-0f887a7c0000` `ee96729a-ada9-71d0-0360-0f887a7c0000`
$OPC_NODES	Returns the names of all regular nodes that are selected at the time the application is executed. The names are separated by spaces. The nodes do not need to be in the node bank. Nodes can be selected directly in a submap of the IP Map. Sample output: `kernighan.c.com richie.c.com`
$OPC_USER	Returns the name of the HPOM user who is currently logged in on the management server. Sample output: `opc_adm`
$OPC_USER_ENCRYPT	Returns the encrypted name of the HPOM user who is currently logged in on the management server.
$OPC_PASSWD	Returns the password of the HPOM user who is currently logged in on the management server.
$OPC_PASSWD_ENCRYPT	Returns the encrypted password of the HPOM user who is currently logged in on the management server.

Variables for Applications Started from the Java GUI

The following variables can only be used in applications started from the Java operator GUI.

$OPC_CUSTOM[*name*]	Returns the value of the custom message attribute name. For example, the variable $OPC_CUSTOM[device] could return the value Lan.
$OPC_EXACT_SELECTED_NODE_LABELS	Returns the labels of all nodes and node groups that are selected at the time the application is executed. The names are separated by spaces.
$OPC_GUI_CLIENT	Executes the application or action on the client where the Java GUI is currently running. This variable is resolved differently, depending on whether the GUI client is running on a UNIX-based system with DNS or on a PC using MS WINS (Windows Internet Name Service). If you are using WINS, $OPC_GUI_CLIENT returns the WINS hostname.
$OPC_GUI_CLIENT_WEB	Starts a web browser on the client where the Java GUI is currently running. This variable is resolved differently depending on whether the GUI client is running on a UNIX-based system with DNS or on a PC using MS WINS (Windows Internet Name Service). If you are using WINS, $OPC_GUI_CLIENT_WEB returns the WINS hostname.
$OPC_NODE_LABELS	Returns the labels of all nodes in the node tree that are selected at the time the application is executed. The names are separated by spaces.

Message-related Variables in the Java GUI

This section describes message-related variables:

- "Parameters for Message-related Variables" on page 113
- "Examples of Message-related Variables" on page 121

Parameters for Message-related Variables

Some variables return the value TRUE or FALSE depending on the existence of a specific message attribute. For example, if an automatic action is defined, TRUE is returned. Otherwise, FALSE is returned.

If an attribute is empty, an empty string is returned. If you use an attribute that does not exist, it is treated like part of a normal string, which means no evaluation happens and the string remains unchanged.

The data returned from variables is exactly the same type as that shown in the Message Properties dialog box.

The indexing for word extraction from strings and for access to specific annotations starts with 1, not with 0.

$OPC_MSG.ACTIONS.AUTOMATIC	Indicates whether or not an automatic action is defined. Sample output: TRUE
$OPC_ MSG.ACTIONS.AUTOMATIC.ACKNOWLEDGE	If an automatic action has been configured to provide an acknowledgement for the selected message, and the actions have been successfully completed, this variable returns yes. Otherwise, no is returned. Sample output: yes
$OPC_ MSG.ACTIONS.AUTOMATIC.ANNOTATION	If this variable returns yes, an automatic action provides annotations for the selected message. If the action fails, an annotation will always be written. Sample output: yes
$OPC_ MSG.ACTIONS.AUTOMATIC.COMMAND	Returns the script or program, including its parameters, performed as an automatic action for the selected message. Sample output: dist_del.sh 30 warning
$OPC_MSG.ACTIONS.AUTOMATIC.NODE	Returns the node on which an automatic action has been performed for the selected message. Sample output: kernighan.c.com
$OPC_MSG.ACTIONS.AUTOMATIC.STATUS	Returns the current status of the message's automatic action. The variable can return running, failed, or successful. Sample output: successful
$OPC_MSG.ACTIONS.OPERATOR	Indicates whether an operator-initiated action is defined. Sample output:

	TRUE
$OPC_ MSG.ACTIONS.OPERATOR.ACKNOWLEDGE	If an operator-initiated action has been configured to provide an acknowledgement for the selected message, and the actions have been successfully completed, this variable returns yes. Otherwise, no is returned. Sample output: yes
$OPC_ MSG.ACTIONS.OPERATOR.ANNOTATION	If this variable returns yes, an operator-initiated action provides annotations for the selected message. Note, if the action fails, an annotation will always be written. Sample output: yes
$OPC_MSG.ACTIONS.OPERATOR.COMMAND	Returns the script or program, including its parameters, performed as an operator-initiated action for the selected message. Sample output: ps -ef
$OPC_MSG.ACTIONS.OPERATOR.COMMAND [*n*]	Returns the *n*th parameter of the script or program, performed as an operator-initiated action for the selected message. Sample output: -ef
$OPC_MSG.ACTIONS.OPERATOR.NODE	Returns the node on which an operator-initiated action has been performed for the selected message. Sample output: kernighan.c.com
$OPC_MSG.ACTIONS.OPERATOR.STATUS	Returns the current status of the message's operator-initiated action. The variable can return running, failed, or successful. Sample output: successful
$OPC_MSG.ACTIONS.TROUBLE_ TICKET.ACKNOWLEDGE	This variable can return the following values:

	yes—The message was automatically acknowledged after having been forwarded to a trouble-ticket system. no—The message was not acknowledged after having been forwarded to a trouble-ticket system. Sample output: yes
$OPC_MSG.ACTIONS.TROUBLE_TICKET.STATUS	This variable can return the following values: yes—The message was forwarded to a trouble-ticket system. no—The message was not forwarded to a trouble-ticket system. Sample output: yes
$OPC_MSG.ANNOTATIONS	Indicates whether or not annotations exist for a message. Returns TRUE if at least one annotation exists for a message. Otherwise, FALSE is returned. Sample output: TRUE
$OPC_MSG.ANNOTATIONS[*n*]	Returns the *n*th annotation. Sample output: Performed Message Correlation; Message Key Relation: Message 59d06840-ac4f-71d5-1f67-0f887e320000 with condition id fe00fa34-9e34-71d5-143e-0f887e320000 ackn'ed 0 messages.
$OPC_MSG.APPLICATION	Returns the name of the application related to the selected message. Sample output: /usr/bin/su(1) Switch User
$OPC_MSG.ATTRIBUTES	This variable can return the following values: unmatched: • Message did not match any message conditions.

	• Message was not originally displayed in the message browser. Sample output: `unmatched`
`$OPC_MSG.CREATED`	Returns the date and time the message was created on the managed node. Sample output: `09/18/08 18:08:08`
`$OPC_MSG.CREATED.EPOCH`	Returns the number of seconds elapsed since January 1, 1970 until the message was created on the managed node. Sample output: `1302619343`
`$OPC_MSG.DUPLICATES`	Returns the number of duplicate messages that have been suppressed. Sample output: `17`
`$OPC_MSG.GROUP`	Returns the message group to which the selected message belongs. Sample output: `Security`
`$OPC_MSG.INSTRUCTIONS`	Returns the text of the instruction. Sample output: `Available space on the device holding the / (root) filesystem is less than the configured threshold. This may lead to ...`
`$OPC_MSG.LAST_RECEIVED`	Returns the date and time when the last duplicate message was received on the management server. Sample output: `09/16/08 03:17:23`
`$OPC_MSG.LAST_RECEIVED.EPOCH`	Returns the number of seconds elapsed since January 1, 1970 until the last duplicate message was received on the

	management server. Sample output: `1302619343`
`$OPC_MSG.MSG_KEY`	Returns the message key that is associated with a message. Sample output: `my_appl_down:kernighan.c.com`
`$OPC_MSG.MSG_ID`	Returns the unique identification number for the selected message. Sample output: `217362f4-ac4f-71d5-13f3-0f887e320000`
`$OPC_MSG.NO_OF_ANNOTATIONS`	Returns the number of annotations of a message. Sample output: `3`
`$OPC_MSG.NODE`	Returns the managed node from which the selected message was issued. Sample output: `kernighan.c.com`
`$OPC_MSG.NODES_INCL_DUPS`	Returns the managed node from which the selected message was issued, including duplicate node names for multiple messages from the same node. Sample output: `kernighan.c.com richie.c.com richie.c.com`
`$OPC_MSG.OBJECT`	Returns the object which was affected by, detected, or caused the event. Sample output: `CPU`
`$OPC_MSG.ORIG_TEXT`	Returns the original text of the selected message. Sample output: `SU 09/18 18:07 + 6 root-spooladm`

`$OPC_MSG.ORIG_TEXT[n]`	Returns the *n*th word in the original text of the message. Sample output: `the`
`$OPC_MSG.OWNER`	Returns the owner of the selected message. Sample output: `opc_op`
`$OPC_MSG.RECEIVED`	Returns the date and time the message was received on the management server. Sample output: `09/18/08 18:08:10`
`$OPC_MSG.RECEIVED.EPOCH`	Returns the number of seconds elapsed since January 1, 1970 until the message was received on the management server. Sample output: `1302619343`
`$OPC_MSG.SERVICE`	Returns the service name that is associated with the message. Sample output: `VP_SM:Agent:ServicesProcesses@@kernighan.c.com`
`$OPC_MSG.SERVICE_LABEL`	Returns the service label that is associated with the message (for example, service label `Disk 3` for service name `node3_disk`). Sample output: `Disk 3` If there is no service label configured, this variable returns an empty string.
`$OPC_MSG.SERVICE.MAPPED_SVC_COUNT`	Returns the number of service names in messages that are mapped to this message. Sample output: `3`
`$OPC_MSG.SERVICE.MAPPED_SVC[n]`	Returns the name of the *n*th service name in this

	message. Sample output: `SAP:applsv01`
`$OPC_MSG.SERVICE.MAPPED_SVCS`	Returns all service names in messages mapped by this message. The names are separated by spaces. Sample output: `SAP:applsv01 SAP:applsv02`
`$OPC_MSG.SEVERITY`	Returns the severity of the message. This can be `Unknown`, `Normal`, `Warning`, `Minor`, `Major`, or `Critical`. Sample output: `Normal`
`$OPC_MSG.SOURCE`	Returns the name of the application or component that generated the message. Sample output: `Message:opcmsg(1|3)`
`$OPC_MSG.TEXT`	Returns the complete text of the selected message. Sample output: `The following configuration information was successfully distributed:` `Templates (OpC30-814)`
`$OPC_MSG.TEXT[`*n*`]`	Returns the *n*th word in the text of the message text. Sample output: `following`
`$OPC_MSG.TIME_CREATED.DAY`	Returns the day when the message was created. Sample output (if the message was created on January 2, 1970 at 10:11:15): `2`
`$OPC_MSG.TIME_CREATED.HOURS`	Returns the hour when the message was created. Sample output (if the message was created on January 2, 1970 at 10:11:15): `10`

$OPC_MSG.TIME_CREATED.MINUTES	Returns the minute when the message was created. Sample output (if the message was created on January 2, 1970 at 10:11:15): 11
$OPC_MSG.TIME_CREATED.MONTH	Returns the month when the message was created. Sample output (if the message was created on January 2, 1970 at 10:11:15): 1
$OPC_MSG.TIME_CREATED.SECONDS	Returns the second when the message was created. Sample output (if the message was created on January 2, 1970 at 10:11:15): 15
$OPC_MSG.TIME_CREATED.YEAR	Returns the year when the message was created. Sample output (if the message was created on January 2, 1970 at 10:11:15): 1970
$OPC_MSG.TIME_OWNED	Returns the date and time when the message was owned. Sample output: 09/18/08 18:11:10
$OPC_MSG.TIME_OWNED.EPOCH	Returns the number of seconds elapsed since January 1, 1970 until the message was owned. Sample output: 1302619343
$OPC_MSG.TYPE	Returns the message type of the message. Sample output: ECS

Examples of Message-related Variables

This section contains examples of messages-related variables and parameters you can use to perform daily tasks.

- Message attributes:

  You can access all message attributes with the following variable:

  `$OPC_MSG.ATTRIBUTES`

  All you would need to do is add an attribute name.

  For example, to get text of a message, you would use the following:

  `$OPC_MSG.TEXT`

  Also when working with attributes that represent strings, you can access a specific word.

  For example, to get the fourth word in the text of a message, you would use the following:

  `$OPC_MSG.TEXT[4]`

  Annotations are an exception to this rule. In annotations, an index specifies the annotation that are returned.

  For example, you would access the seventh annotation of the current selected messages with the following:

  `$OPC_MSG.ANNOTATIONS[7]`

- Duplicate messages:

  If you need to find information about the number of message duplicates for an application, use the following:

  `$OPC_MSG.DUPLICATES`

- Creation time and severity:

  If want to do some use time and severity to do statistical calculations, you would specify the message creation time and the severity, as follows:

  `$OPC_MSG.CREATED`

  `$OPC_MSG.SEVERITY`

- Message text:

  If you have defined a policy condition that creates a message text with some status as the third word, and you would like to extract this status easily and forward it to an application called `evaluate_status`, you would use the following:

  `evaluate_status $OPC_MSG.TEXT[3]`

- Action attributes:

  If you want to use and evaluate action attributes, you can write shell scripts that check for automatic and operator-initiated actions, and get more information about the action status and if they are annotated:

  `script_name` `$OPC_MSG.ACTIONS.AUTOMATIC`

  `script_name` `$OPC_MSG.ACTIONS.AUTOMATIC.STATUS`

  `script_name` `$OPC_MSG.ACTIONS.AUTOMATIC.ANNOTATION`

The first parameter would be `TRUE` if an automatic action was defined for the message. This script would be useful only if there are more attributes used afterwards, but not to check for every attribute if it is an empty string.

- Annotations:

  To access the second annotation of a selected message in an application, you would use the following:

  ```
  $OPC_MSG.ANNOTATIONS[2]
  ```

Variables Used with Service Navigator

This section lists and defines the variables that can be used with Service Navigator.

HPOM Variables in Service Names

HPOM enables you to use variables as part of the service name in the `Service Name` field.

For example, consider the following services:

```
SAP:applsv01
```

```
SAP:applsv02
```

To intercept messages from all application servers, enter `SAP:<$MSG_NODE_NAME>` in the `Service Name` field.

The message condition resolves the variable to `applsv01` or `applsv02` depending on where the message originated from, and sends it to the corresponding service.

For more information about variables that can be used in the `Service Name` field, see "Variables in Message Source Policies" on page 100. Note that you cannot use these variables in the service configuration file.

HPOM Variables in Tools and Service Actions

The following variables can be used in the command string when defining a service action in the service configuration file:

`$OPC_SERVICE_LABEL`	Returns the label of a service.
`$OPC_SERVICE_MAPPINGS_SVC_COUNT`	Returns the number of “service name in message” properties of a service. Sample output: `17`

$OPC_SERVICE_MAPPINGS_SVC[*n*]	Returns the *n*th “service name in message” of a service. Sample output: SAP:applsv02
$OPC_SERVICE_MAPPINGS_SVCS	Returns all “service name in message” properties of a service. The names are separated by spaces. Sample output: SAP:applsv01 SAP:applsv02
$OPC_SERVICE_NAME	Returns the name of the current service.
$OPC_SERVICE_NODE	Returns the name of the node service attribute, if set.
$OPC_SERVICE_ORIGINAL_ID	Returns the original ID of a service. Sample output: node_fred
$OPC_SERVICE_VALUE [name]	Returns the value of the service parameter with name <name> for the selected service; returns an empty string if not set.

Note: All these HPOM variables can also be used with tools.

HPOM Variables in URL Definitions

The following variables can be used when defining a URL:

$LANG	Returns the language setting of the user who is running a Service Navigator GUI.
$OPC_GUI_CLIENT	Returns the hostname of the client the GUI is currently running on.
$OPC_MGMTSV	Returns the hostname of the HP Operations management server.

Chapter 3: Configuring HPOM Managed Nodes

In this Chapter

This chapter contains details about HPOM managed nodes, such as configuration and reference information, server-node communication, and so on. The information in this chapter covers the following topics:

For more detailed information about HPOM managed nodes, see the *HPOM Concepts Guide* and the HP Operations Agent documentation.

HTTPS Communication Administration Commands in HPOM

HTTPS Communication can be controlled using the following commands:

- On the management server and managed nodes:
  - **ovcoreid** (Unique System Identifier)

    The `ovcoreid` command is used to display existing `OvCoreId` value and, in addition, create and set new `OvCoreId` values on the local system.

For details of how to use this tool, see the *ovcoreid(1)* manual page.

- **ovc** (Process Control)

  `ovc` controls starting and stopping, event notification, and status reporting of all components registered with the Control service, `ovcd`. A component can be a server process, an agent (for example, the Discovery Agent), an event interceptor, or an application delivered by an integrator.

  For details of how to use this tool, see the *ovc(1)* manual page.

- **bbcutil**

  The `bbcutil` command is used to control the HP Communication Broker.

  For syntax information and details of how to use this tool, see the *bbcutil(1)* manual page.

  Communication parameters are set in the *<OVDataDir>*`/conf/confpar/bbc.ini` file.

- **ovconfget**

  Installed HP BTO Software components have associated configuration settings files that contain one or more namespaces and apply system wide or for a specified High Availability Resource Group. A namespace is a group of configuration settings that belong to a component. All configurations specified in the settings files are duplicated in the `settings.dat` configuration database.

  For each specified namespace, `ovconfget` returns the specified attribute or attributes and writes them to `stdout`. Used without arguments, `ovconfget` writes all attributes in all namespaces to `stdout`.

  For details of how to use this tool, see the *ovconfget(1)* manual page.

- **ovconfchg**

  Installed HP BTO Software components have associated configuration settings files that contain one or more namespaces. A namespace is a group of configuration settings that belong to a component.

  `ovconfchg` manipulates the settings in either the system-wide configuration file or the configuration file for the specified `High Availability Resource Group`, updates the configuration database, and triggers notification scripts.

  For details of how to use this tool, see the *ovconfchg(1)* manual page.

- **ovpolicy**

  `ovpolicy` manages local policies and policies. A policy is a set of one or more specifications, rules and other information that help automate network, system, service, and process management. Policies can be deployed to managed systems, providing consistent, automated administration across the network. Policies can be grouped into categories. Each category can

have one or more policies. Each category can also have one or more attributes, an attribute being a name value pair.

You use `ovpolicy` to install, remove, enable, and disable local policies. For details of how to use this tool, see the *ovpolicy(1)* manual page.

- On the HP Operations management server:
  - **opccsacm** (Certificate Server Adapter Control Manager)

    The `opccsacm` command is used to issue new node certificates and installation keys manually on the HP Operations server. It also modifies the HP Operations database to reflect the changes made by certificate management actions.

    For details of how to use this tool, see the *opccsacm(1m)* manual page.

  - **opccsa** (Certificate Server Adapter)

    The `opccsa` command is used to list the pending certificate requests, map certificate requests to target nodes from the HP Operations database, grant, deny and delete specified certificate requests.

    For details of how to use this tool, see the *opccsa(1m)* manual page.

For commands used only on HPOM managed nodes, see the HP Operations agent documentation.

Remote Action Authorization

Action requests contained in HPOM messages that specify a target system for the action (other than the sender of the message) are remote actions and must be handled securely. The execution of such actions can be controlled by using the `remactconf.xml` file, which you can find in the following location:

`/etc/opt/OV/share/conf/OpC/mgmt_sv/remactconf.xml`

For more information about remote actions, see "Remote Actions" on page 588 and the *HPOM Concepts Guide*.

Management Server Configuration for Remote Action Authorization

The message manager uses a file-based configuration on the HP Operations management server to specify authorization of remote actions. For detailed information about the remote action configuration file syntax, see "Remote Action Configuration File Syntax (XML-based)" on page 129.

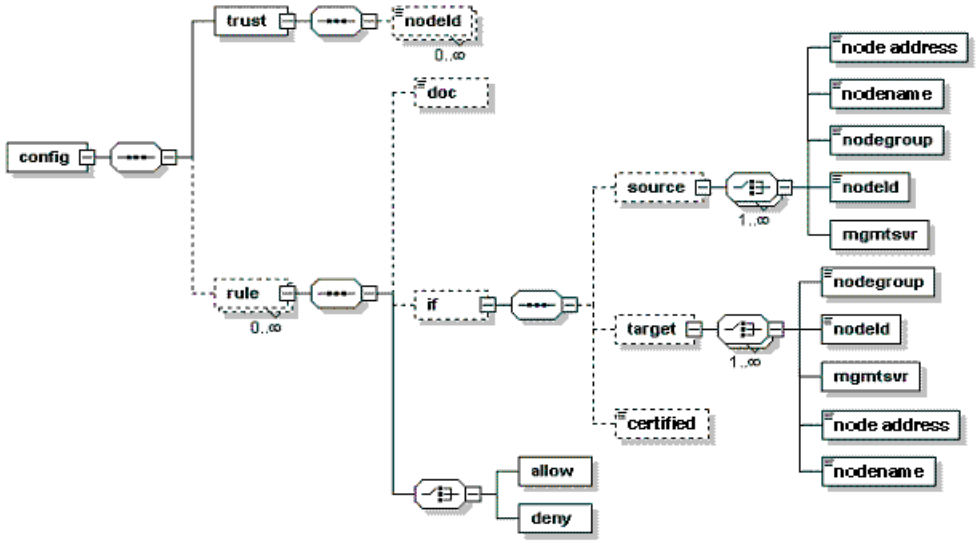
When configuring the management server for remote action authorization, the following general rules apply:

- When a message containing a remote action arrives on the HP Operations management server, this file is reloaded if modified and the message is processed according to the rules contained in the remote action configuration file.
- If the remote action configuration file does not exist or if it is empty, unreadable, or does not contain rules, all remote actions are disabled. If it contains syntax errors or other logical errors, such as a non-existing node group, parsing stops and all subsequent rules are ignored.
- A message containing remote actions is matched against the rules in the same sequence. The first match determines the result: the `deny` clause disables remote actions within the message and adds an appropriate annotation to the message, while the `allow` clause leaves the message unmodified.
- If the message does not match any rule, remote actions are disabled in the same way as if the message matched the `deny` rule.
- A rule matches if all rule elements match in an AND logic. If a possible rule element is omitted (for example, no `target` tag is specified), any appropriate message value matches. However, this does not apply to the `certified` tag—if it is not specified, the default of `true` applies.
- The `trust` section is not supported.
- The `certified` tag has the `true` (default) value. The message must originate from a certified source and the message certificate must be verified. A rule containing the `<certified>true</certified>` clause matches messages from managed nodes.
- Regular or MSI-created (not modified) actions from the managed node can be allowed by setting the following condition as part of a rule in the `remactconf.xml` file:

  ```
  <if>
    <certified>msi</certified>
  </if>
  ```

- Authorization data is logged with the reason for denying authorization. If an action is unauthorized, it is automatically deleted from the message and details about the match and the signature status are added as an annotation to the message. Unauthorized messages never appear in the GUI and therefore cannot be accidentally executed.
- Source and target nodes are matched against node groups or single nodes. A dedicated keyword can be used for the management server.

Figure 1: Remote Action Configuration File Syntax (XML-based)



The remote action configuration file contains the following components:

`config`	Consists of the `trust` element that defines which systems are trusted as action signers and a list of `rule` elements.
`trust`	Consists of a list of `nodeId` elements, each containing the `OvCoreId` of a trusted node.
`rule`	Consists of the following components: • `doc` (*optional*) • `if` (*optional*) containing `condition` • `allow` or `deny` action The `allow` and `deny` actions are empty and define if an action request is authorized (allowed) or rejected (denied).
`condition`	Consists of optional checks (`source`, `target`, and `certified`). A condition matches only if all contained checks match. If no check is defined or if no condition is defined, a match is always successful. • `source` (used to check the source node of an action request) and `target` (used to check the target node of an action request) consist of the following: • `nodegroup` (contains the name of a node group from the HPOM database—it matches if the request's node is a member of that node group). • `nodeId` (contains the `OvCoreId`—it matches if this `OvCoreId` is the ID of the request's node).

	• `mgmtsrv` (empty element—it matches if the request's node is the management server). • `nodeAddress` • `nodename` • `certified` check consists of the following values: • `valid` (matches only if a signature and a certificate are provided—the signature must be signed by the certificate's owner and the `OvCoreId` of the certificate's subject must be listed in the `trust` element) • `invalid` (matches cases that are not described under `valid`)

The following is an example of a remote action configuration:

```
<?xml version="1.0"?>
<config xmlns="http://openview.hp.com/xmlns/Act/Config/2002/08">
  <rule>
    <doc>Actions from Group2 to Group1 are always allowed</doc>
    <if>
      <source>
        <nodegroup>Group2</nodegroup>
      </source>
      <target>
      <nodegroup>Group1</nodegroup>
      </target>
    </if>
    <allow/>
  </rule>
  <rule>
    <doc>No actions from Group3 are allowed</doc>
    <if>
      <source>
        <nodegroup>Group3</nodegroup>
      </source>
    </if>
    <deny/>
  </rule>
  <rule>
    <doc>Actions to Group3 are allowed if certified</doc>
    <if>
      <target>
        <nodegroup>Group3</nodegroup>
      </target>
      <certified>true</certified>
    </if>
    <allow/>
```

```
  </rule>
</config>
```

When specifying a pattern for a node name or the IP address of a node in the remote action configuration file, keep in mind that the syntax of the remote action configuration file is XML-based. The less than (<) and greater than (>) signs that you need for HPOM pattern matching are special characters in XML and therefore must be escaped—instead of "<" use "<" and instead of ">" ">".

The pattern matching used is the HPOM pattern matching. For example, the pattern `<*>.rose.hp.com` with XML escape sequences looks as follows:

```
<rule>
 <doc>Allow *.rose.hp.com -> *.rose.hp.com certified actions</doc>
 <if>
  <source>
   <nodename>&lt;*&gt;.rose.hp.com$</nodename>
  </source>
  <target>
   <nodename>&lt;*&gt;.rose.hp.com$</nodename>
  </target>
 </if>
 <allow/>
</rule>
```

Roles and Access Rights

In general, a role grants the right to perform a certain task; for example, in HPOM environments, the rights to execute actions, deploy files, or configure settings. Each preconfigured HPOM role described below has a default set of access rights that can be changed as explained in the HP Operations agent documentation.

Access rights are the rights to, for example, execute actions, deploy files, and configure settings. The rights are mapped to the HP Operations management server roles described in the *HPOM Concepts Guide*.

It is possible to alter the mappings by changing configuration settings under the namespace `sec.core.auth.mapping.<HPOM_mgr_role>`, where *<HPOM_mgr_role>* is the role of the HP Operations management server. For example, to avoid accidental or unauthorized configuration deployment, you may want to disallow policy and instrumentation deployment from the initial HP Operations management server.

Restricting Access Rights

You can restrict access from the HP Operations management server processes to the HTTPS agents and thereby limit or disallow the operations a management server can perform on a managed node.

You can grant specific access rights either locally on each individual HTTPS managed node using the `ovconfchg` command-line tool (for more information, see the HP Operations Agent documentation), or remotely from the HP Operations management server at agent installation time, by adding the required settings to the `bbc_inst_defaults` file.

Tip: If you add the settings to the `bbc_inst_defaults` file, you do not need to change settings on individual HTTPS agents. You can limit these settings to subnets, individual nodes, and so on within the `bbc_inst_defaults` file.

For more information about two common scenarios, see also "Avoiding Unattended Configuration Deployment" on page 133 and "Denying Remote Access" on page 134.

When you use the `bbc_inst_defaults` file to change access rights, you must replace the following variables with one of the possible values listed below:

Variable	Description and values
`sec.core.auth.mapping.` *`<HPOM_mgr_role>`*	Namespace of the initial HP Operations manager: `sec.core.auth.mapping.manager` Flexible management environments only: `sec.core.auth.mapping.secondary` `sec.core.auth.mapping.actionallow` For more information about each role, see the *HPOM Concepts Guide*.
`<comp_name>`	Agent component names: `ctrl` `conf` `depl` `eaagt.actr`
`<dec_value>`	Sum of the decimal values representing the access rights of an HP Operations manager for a particular agent component. The default values are: • `ctrl`: 15 • `conf`: 511

	• depl: 2047 • eaagt.actr: 1 For a detailed list of access rights and their corresponding values, see the *HPOM Concepts Guide*.

To restrict access to HTTPS agents remotely from the management server at installation time, specify the desired settings in the bbc_inst_defaults file:

/etc/opt/OV/share/conf/OpC/mgmt_sv/bbc_inst_defaults

```
[sec.core.auth.mapping.<HPOM_mgr_role>]
   <comp_name> = <dec_value>
   <comp_name> = <dec_value>
   ...
```

Avoiding Unattended Configuration Deployment

To avoid unattended configuration deployment, you can deny configuration deployment from the HP Operations management servers by setting the following values for one or more of the HP Operations manager roles:

conf	496
depl	2044

For example, use the ovconfchg command line tool on a managed node to deny configuration deployment from the initial HP Operations manager, enter:

```
ovconfchg -ns sec.core.auth.mapping.manager -set conf 496 -set depl 2044
ovc -kill
ovc -start
```

You can also deny configuration deployment from the initial HP Operations management server to all nodes within a specified subnet (192.168.10 in the following example) so that only authorized experts can update these security-sensitive nodes locally. Add the following lines to the bbc_inst_defaults file before installing the nodes:

```
[sec.core.auth.mapping.manager]
   192.168.10.* : conf = 496
   192.168.10.* : depl = 2044
```

An error message is generated when a configuration distribution request is triggered accidentally (or without authorization) on the management server.

Denying Remote Access

To completely deny remote access to an HP Operations agent, set the following values for one or more of the HP Operations manager roles:

ctrl	0
conf	256
depl	0
eaagt.actr	0

For example, run the following commands locally on a managed node:

```
ovconfchg -ns sec.core.auth.mapping.manager -set ctrl 0 -set conf 256 -set depl 0 -
set eaagt.actr 0
```

```
ovc -kill
```

```
ovc -start
```

You can also add the following lines to the `bbc_inst_defaults` file before installing the nodes:

```
[sec.core.auth.mapping.manager]
   192.168.10.<*> : ctrl = 0
   192.168.10.<*> : conf = 256
   192.168.10.<*> : depl = 0
   192.168.10.<*> : eaagt.actr = 0
```

The management server will still be able to receive messages from the managed node but will not be able to access the node from a remote location. To revert this setting, use the `ovconfchg` command line tool locally on the managed node.

Working with Certificates

Certificates are needed for network communication using the Secure Socket Layer (SSL) protocol with encryption. Server and client authentication are enabled. Managed nodes of the managed environment are identified using certificates. The “SSL handshake” between two managed nodes only succeeds if the issuing authority of the certificate presented by the incoming managed node is a trusted authority of the receiving managed node.

For more information on certificates, see the *HPOM Concepts Guide* and the HP Operations agent documentation.

You can install certificates automatically and manually. See the following sections:

- "Deploying Certificates Automatically" on page 135.
- "Generating Certificate for Manual Certificate Deployment" on page 142.
- "Deploying Manual Certificate with an Installation Key" on page 145.
- "Displaying Certificate States" on page 145.

Node Information

For detailed information about the node, enter the following command:

```
opccsa -list_pending_cr –format rhiomp
```

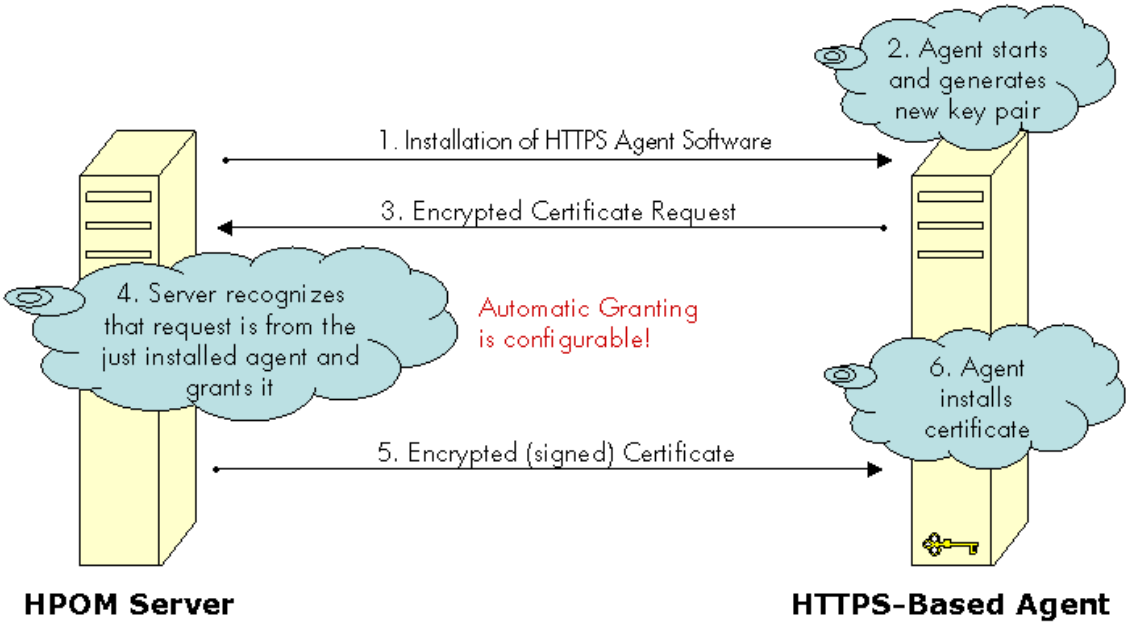
In this instance, `rhiomp` stands for:

h	Hostname: the hostname of the node that initiated the certificate request (not a unique identifier).
i	IP address: the IP address of the node that initiated the certificate request (not a unique identifier).
o	`OVCoreID`: the only unique identifier of an HPOM HTTPS node. When you grant a request, you also grant all communication originating from the node with this `OvCoreID`. The hostnames can be changed, but the `OVCoreID` remains the unique identifier of the node.
m	Mapped to: the hostname of the node to which listed certificate requests are mapped.
p	Platform: the operating system of the HPOM managed node.

Deploying Certificates Automatically

The most common certificate deployment method is to let HPOM create, grant and distribute certificates automatically. Figure 2 illustrates how HPOM issues certificates to HTTPS managed nodes.

Figure 2: Certificate Deployment Process



After the HTTPS agent software is installed on a managed node system, the certificate management client on the node system creates a private key and a certificate request. A secret key is used to encrypt the certificate request which is sent over the network to the server system. Automatic granting is the default configuration and the autogrant interval is set to 30 minutes. If a request arrives after the allowed time interval, it must be handled by using the `ovcm -grant` command. If you wish to change this interval, use the following command:

```
ovconfchg -ovrg server -ns opc -set OPC_CSA_AUTOGRANT_INTERVAL <time interval in minutes>
```

If the message is encrypted with the correct key, the receiving management server trusts the sender. This does not provide full security, and is not recommended for highly secure environments but is more secure than transmitting the requests as plain text. This mode is only used for transmitting the certificate request and the signed certificate, which should be a short period of time.

In secure environments, it is recommended that automatic granting of certificate requests is disabled and that an administrator assesses each request before granting those that are valid. You can do this with the command:

```
ovconfchg -ovrg server -ns opc -set OPC_CSA_USE_AUTOGRANT <TRUE|FALSE>
```

However, manual installation of certificates is the only fully secure method.

Note: A secret key is part of the HTTPS security software and is used by default for all HP Operations HTTPS-based applications. Every installation uses the same secret key.

A configurable secret key is a user configured key that replaces the secret key. This can be done before the management environment is setup. Ensure that every system that may request a certificate is using the same secret key as the certificate server.

Using a configured secret key ensures that a client system is not able to request a certificate from a foreign certificate server system, for example another HP Operations installation.

Note: The Certificate Server system must be setup and active before certificates can be generated and distributed.

To automatically deploy certificates, install the HTTPS agent software on a managed node system. After the installation, the following steps are executed by HPOM:

1. A new public/private key pair is generated on the managed node system by the certificate management client.
2. The managed node system initiates a certificate request on the node system.
3. The generated private key is stored in an encrypted file.
4. The certificate request is encrypted with the secret key and sent to the Certificate Server system (using a non-SSL connection as the node system does not yet have a valid certificate).
5. After the certificate request has been decrypted successfully on the Certificate Server it is added to the pool of pending certificate requests and a notification is sent to all registered components, and corresponding entry in the HPOM Event Browser is also displayed.
6. The certificate request is either granted or denied by matching certain preconfigured criteria. For example, the request was made within 2 minutes of the HTTPS agent software being installed on the node system.

   Note: Granting of a certificate request is the most security sensitive step in this process. The instance that grants the request should have a good reason to do this. An example would be an administrator who is waiting for a request after deploying a package to the node that now requests a certificate from the certificate server.

7. If the request is granted, the certificate request is signed by the Certificate Server. The signed certificate is then encrypted with the secret key and sent to the node system.

   If the certificate request is denied, the server system sends a message to the node system indicating that the request has been rejected and corresponding entry in the HPOM Event Browser is also displayed.
8. The Certificate Client on the node system receives the response. If the request has been granted, it installs the new certificate and is now ready to use SSL for authenticated connections.

   If the certificate request has been denied, the Certificate Client stores this information to prevent an automatic retry.

Automatic Processing of Certificate Requests

To enable or disable automatic processing of certificate requests from HP Operations agents and allow the automatic adding of systems to the HPOM node bank before granting a certificate request, set the OPC_CSA_AUTOMATION server configuration variable to TRUE or FALSE. When specifying a list of rules and subnet patterns for automatic certificate processing, use the OPC_CSA_RULES server configuration variable or the OPC_CSA_NAT_RULES server configuration variable (in a NAT environment).

For more information about the server configuration variables, see the *HPOM Server Configuration Variables* document.

When specifying a list of rules for automatic certificate processing, make sure that rules are valid. A rule is valid if it has at least one of the following tasks:

- DENY_ACTION
- PRE_ACTION
- GRANT
- DENY
- DELETE

The POST_ACTION and ADD_NODE tasks are optional. ADD_NODE is valid only with GRANT and ignored with DENY and DELETE.

Caution: The order in which the tasks are executed is always as follows: DENY_ACTION, PRE_ACTION, ADD_NODE, GRANT|DENY|DELETE, POST_ACTION.

Each rule has the following form:

```
<rule_name>=[DENY_ACTION:<deny_action_script>,]
[PRE_ACTION:<pre_action_script>,][ADD_NODE,]
GRANT|DENY|DELETE[,POST_ACTION:<post_action_script>]
```

For example:

```
OPC_CSA_RULES=rule1+(*.mydomain.com);rule2+(*)
rule1=PRE_ACTION:/tmp/precsad.sh,GRANT,ADD_NODE, POST_ACTION:/tmp/postcsad.sh
rule2=DENY,POST_ACTION:/tmp/csaddeny.sh
```

In this example, rule1 is applied on all accepted certificates matching *.mydomain.com, whereas rule2 is used to deny undesired certificate requests.

For subsequent tasks to be executed, each rule has to be valid. This means that the PRE_ACTION task must have a return code that is equal to zero. When OPC_CSA_NAT_RULES is used for specifying a list of rules and subnet patterns for automatic certificate processing in a NAT environment, the node must also be resolvable (that is, the node name, the IP address, or both must be resolvable). If the node

name and the IP address are not recognized on the management server, the PRE_ACTION task must return both.

The PRE_ACTION script outputs the following values in the key=value format:

- IPAddress
- Nodename
- Nodegroup
- Label

Note: The certificate is not auto-granted unless the node is already in the node bank or the ADD_NODE task is specified as part of the rule.

A managed node is searched for by using an OvCoreId. If the managed node is not found by using the OvCoreId, it is searched for by using a hostname. In this case, if the PRE_ACTION script returns a hostname, the hostname is checked. If the PRE_ACTION script does not return a hostname, the hostname from the certificate request is checked.

If the managed node is found by using a hostname, the IP address from the PRE_ACTION script or the certificate request is compared with the IP address from the database. If they differ, the OPC_CSA_ALLOW_IP_MISMATCH setting is checked. If the OvCoreId in the database is not empty (that is, if it differs from the one in the certificate request), auto-mapping is not executed.

Caution: The communication type of the managed node in the database must be BBC regardless of how the managed node is found, by using the OvCoreId or the hostname.

Automatic Denial of Certificate Requests from an HP Operations Agent

You can use CSA automation to deny certificate requests from an HP Operations agent based on its version or platform. This means that after you add an HP Operations agent version and/or platform to a list of unwanted items, every certificate request coming from such a managed node is denied automatically. To do so, follow these steps:

1. *Prerequisites:* Before enabling this feature, make sure your system meets the following prerequisites:
   - HP Operations agent version 12.x is installed on the HP Operations management server and the Core patch is reapplied.
   - The managed node already exists in the database.

     Note: This is important because the HP Operations management server requires the information about the HP Operations agent version and platform to store it in the database.

The information is used by the script for determining whether a certificate request should be denied or not.

- Automatic certificate granting is disabled.

  If it is not disabled, run the following command:

  ```
  ovconfchg -ovrg server -ns opc -set OPC_CSA_USE_AUTOGRANT FALSE
  ```

2. Enable automatic certificate request processing (that is, CSA automation) by running the following command:

   ```
   ovconfchg -ovrg server -ns opc -set OPC_CSA_AUTOMATION TRUE
   ```

3. Make sure the script can access the HP Operations agent-related data by running the following command:

   ```
   ovconfchg -ovrg server -ns opc -set OPC_CSA_ADD_OS_OA_DATA_IN_AUTOMATION TRUE
   ```

4. Add a CSA rule that runs the `opc_csa_deny_agt_os_ver.pl` script as a `DENY_ACTION` by running the following commands:

   ```
   ovconfchg -ovrg server -ns opc.opccsad -set OPC_CSA_RULES "denyrule+(*)"
   ovconfchg -ovrg server -ns opc.opccsad -set denyrule DENY_
   ACTION:/opt/OV/contrib/OpC/autogranting/opc_csa_deny_agt_os_ver.pl
   ```

   Caution: If you were already using CSA automation, make sure to modify these commands appropriately because they overwrite the existing CSA rules.

5. Configure rules so that the script can determine whether a certain certificate request should be denied or not. To do so, use the `OPC_CSA_INCLUDE_OS_OA_CONFIG` server configuration variable as follows:

   ```
   ovconfchg -ovrg server -ns opc -set OPC_CSA_INCLUDE_OS_OA_CONFIG rules
   ```

   When configuring the rules, keep in mind the following:

   - If you have more than one rule, the rules must be separated by slash marks. For example:

     ```
     rule1/rule2/.../ruleN
     ```

   - Each rule consists of rule segments and it must have the following format:

     ```
     OS family:OS type:OS name:OS vendor:OS version:Agent version:Exit code
     ```

     If any of the rule segments is omitted, the rule segment matches any item. For example, an empty `OS vendor` rule segment, which is indicated by `::`, matches any OS vendor. Any rule with an invalid exit code (that is, a non-integer number) is ignored.

Table 8: Rule Segments and Their Explanations

Rule Segment Name	Explanation
`OS family`	Possible values are `unix` and `windows`.
`OS type`	For example: `Linux` TIP: This is the output of the `uname -s` command on UNIX platforms.
`OS name`	For example: `CentOS 6.5`
`OS vendor`	For example: `Hewlett Packard`
`OS version`	For example: `2.6.18`
`Agent version`	HP Operations agent version (for example, `11.11.035`)
`Exit code`	◦ `0`: Do not deny a request. ◦ `1`: Deny a request without a specific reason. ◦ `4`: Deny a request due to an unwanted OS version. ◦ `5`: Deny a request due to an unwanted HP Operations agent version.

Note: The rules are evaluated one by one in the order in which they are listed. When a matching rule is found, no further rules are evaluated.

The following is a list of examples that shows how to set a correct rule format when you want to do one of the following:

- Allow all certificate requests from Windows nodes regardless of the HP Operations agent and OS versions:

  ```
  windows::::::0
  ```

- Deny certificate requests from CentOS 5.x nodes:

  ```
  ::CentOS 5::::4
  ```

- Deny certificate requests from HP Operations agents 11.00.xxx:

  ```
  :::::11.00:5
  ```

  To combine all three example rules, run the following command:

  ```
  ovconfchg -ovrg server -ns opc -set OPC_CSA_INCLUDE_OS_OA_CONFIG
  'windows::::::0/::CentOS 5::::4/:::::11.00:5'
  ```

Generating Certificate for Manual Certificate Deployment

Certificates can be deployed totally manually. This avoids sending any certificate-related information over the network before SSL communication is established. The public/private key pair is generated on the certificate server and then transported to the managed node system. This method is often chosen for highly secure environments where it is undesirable to transmit certificate and key data over a network.

Note: The Certificate Server system must be setup and active before certificates can be generated and distributed.

To manually deploy certificates that have been generated on the Certificate Server:

1. If you are dealing with a particularly large environment, you can create the `bbc_inst_defaults` file to maintain common settings for managed node on the HP Operations management server. The file should be located as follows:

   `/etc/opt/OV/share/conf/OpC/mgmt_sv/bbc_inst_defaults`

   In the namespace `sec.cm.client`, set the deployment type for your managed nodes to manual by adding an entry of the following type for each managed node:

   <IP address>`: CERTIFICATE_DEPLOYMENT_TYPE = MANUAL`

   For example:

   `192.168.10.17: CERTIFICATE_DEPLOYMENT_TYPE = MANUAL`

   The IP address can accept wildcards to specify ranges of managed node.

   For further information, see the following file:

   `/etc/opt/OV/share/conf/OpC/mgmt_sv/bbc_inst_defaults.sampl`

   See the HP Operations agent documentation for some examples of how to use the `bbc_inst_defaults` file.
2. If installing the HTTPS agent software manually, create a default profile as described in the HP Operations agent documentation.
3. Install the HTTPS agent software on the selected managed node system, manually or remotely.
4. Make a note of the `OvCoreId` value assigned to the selected managed node. `OvCoreId` can be retrieved by calling one of the following commands:
   - `ovcoreid`
   - `ovconfget sec.core`

When an agent is newly installed by using the `inst.sh` script, a new OvCoreId is created. However, if an OvCoreId is already present in the HPOM database for the managed node system, this is used in preference.

When installing the agent software manually, you must create a profile, copy it and the software packages to the managed node system. The profile includes the original OvCoreId from the HPOM database. Install the profile by running the following command:

```
oainstall.sh -configure -agent -agent_profile <profile>
```

Note: The `OvCoreId` stored on a remote system can be determined by using the command:

```
bbcutil -ping http://<remote system>
```

provided that the Communication Broker is running on the remote system.

The `OvCoreId` can also be locally displayed with the `ovcoreid` command.

The `OvCoreId` value stored for the managed node in the HPOM database can be displayed with the command:

```
opcnode -list_id node_list=<nodename>
```

5. On the HP Operations management server system, ensure that the selected managed node is added to the node bank.
6. As an HPOM administrator, create a signed certificate and the corresponding private key for a specific managed node manually on the Certificate Server system using the `opccsacm` command line tool. You must provide a password to encrypt the created data.

   Note: If certificates must be created before the HTTPS agent software is installed on the selected managed node, it is possible to specify the `OvCoreId` (`coreid` parameter) in the following command. A `OvCoreId` is still created and it is stored in the database.

   The `OvCoreId`, which is part of the certificate file name, can be retrieved with the command if the managed node is already stored in the HPOM database:

   ```
   opcnode -list_id node_list=<node name>
   ```

   This value must then be set on the corresponding node system after the HTTPS agent software is installed with the command:

   ```
   ovcoreid -set <id> -force
   ```

   If no `OvCoreId` is already stored, use the value from the managed node:

   The `OvCoreId` stored on a remote system can be determined by using the command:

   ```
   bbcutil -ping http://<remote system>
   ```

   provided that the Communication Broker is running on the remote system.

Alternatively, the `OvCoreId` can be displayed locally by using the `ovcoreid` command.

To create a certificate for the selected managed node, on the HP Operations management server system, enter the command:

```
opccsacm -issue -file <filename> [-pass <password>] -name <full_qual_hostname>
-coreid <OvCoreId>
```

The tool asks you to specify a password to encrypt the created certificate. This is later required to decrypt the certificate when importing the certificate to the managed node system.

7. Set the installation type to `MANUAL`, either in the `bbc_inst_defaults` file or with the command:

   ```
   ovconfchg -nssec.cm.client -set CERTIFICATE_DEPLOYMENT_TYPE MANUAL
   ```

   Copy the file containing the signed certificate, its corresponding private key and the root certificate onto a floppy disk or other portable media.

   The default file location directory if the `-file` option was omitted is:

   ```
   /<OvDataDir>/temp/OpC/certificates
   ```

   The file name takes the following form:

   ```
   <hostname>-OvCoreId.p12
   ```

8. Go to the managed node system and stop the agent locally with the command:

   ```
   ovc -stop
   ```

9. Install the certificate, the trusted root certificates and the private key from the portable media using the `ovcert` command line tool. Specify the password used in step 6 when requested during installation of the certificate.

   To import the certificate, enter the following command:

   ```
   ovcert -importcert -file <file created in step 6>
   ```

   The tool will ask for the password that was provided in step 6.

   Note: Access to the medium that contains private keys should be tightly controlled to ensure that only authorized people can use them.

10. After installation, delete the certificate installation file from the managed node, and delete the data on the portable medium or store it in a secured place.
11. Start the agent locally with the command:

    ```
    ovc -start
    ```

12. Delete the file created for the certificate import from the certificate server system.

Deploying Manual Certificate with an Installation Key

Manual certificate deployment with an installation key offers the advantage that the private key never leaves the system to which it belongs. However, it requires that some security-related data is transmitted over the network before the certificate can be installed on the managed node system.

Note: The Certificate Server system must be setup and active before certificates can be generated and distributed.

To manually deploy certificates using an installation key:

1. Manually install the HTTPS agent software on the managed node system. For further information, see the HP Operations agent documentation.
2. As an HPOM administrator, initiate the creation of a new installation key on the Certificate Server system. Provide a password to encrypt the created key.

   `opccsacm -geninstkey -file <filename> [-pass <password>]`

   The Certificate Server adds the key to its installation key repository and writes it, together with some management information to a file.
3. Copy the file with the installation key information onto a floppy disk or other portable media.
4. Go to the managed node system and, using the `ovcert` command line tool, initiate a new certificate request. A new public/private key pair is generated. Use the following command:

   `ovcert -certreq -instkey <filename>`

   The encrypted request is sent to the Certificate Server.

The Certificate Server decrypts the request with the key from its repository.

If the correct installation key was used, the Certificate Server automatically grants the request and sends the signed certificate back to the managed node. Then it removes the installation key from the repository. If an invalid installation key was used, the request is automatically denied.

Displaying Certificate States

To display the certificate states of nodes, use this Certificate State Report:

`/opt/OV/bin/OpC/call_sqlplus.sh cert_state`

Certificate States Overview

These are two different scenarios that might happen with node certificate states.

Depending on the actions, the certificate states change as follows:

Table 9: Certificate States Workflow Scenario 1

Certificate State	Action	Description
NO	A node is added to the node bank.	There is no agent installed. The certificate was not requested yet.
PENDING	The agent is installed manually.	The certificate request is granted, but the certificate is not yet installed on the agent. The certificate is not granted yet. The agent is installed and activated and therefore the certificate server got certificate request from it.
GRANTED	The certificate request is granted.	The certificate request is granted, but the certificate is not yet installed on the agent. Once the certificate request is granted, the certificate is installed automatically on the agent. The state is changed to YES. Note that the state might not be visible.
GRANTED	The agents are stopped with ovc -kill.	The certificate state remains granted only if the managed node is unreachable. For example the agents were stopped. The certificate server tries to send the certificate to an unreachable managed node every minute after it is granted. After the time limit of two hours, the certificate request is removed from the queue. This means a new certificate request needs to be created on the managed node with ovcert -certreq.
YES	The certificate is installed.	The certificate is installed on agent.

Table 10: Certificate States Workflow Scenario 2

Certificate State	Action	Description
PENDING	The agent is activated and thus the certificate is requested.	The certificate request is PENDING.
PENDING	The agent processes are stopped using ovc -kill.	Since the processes are not running, the certificate cannot be delivered or installed. The certificate state remains PENDING in the GUI, it is not GRANTED yet.

Certificate States Workflow Scenario 2 , continued

Certificate State	Action	Description
GRANTED	The certificate is granted in the GUI.	Now the certificate is GRANTED in the GUI, but since the Core processes on the agent are not running, the certificate cannot be delivered and installed yet. Therefore the state remains GRANTED.
GRANTED	The agents are started with `ovc -start`.	The certificate is requested. The state is still GRANTED. The certificate state remains GRANTED until certificate server tries to install the certificate again.
YES	The certificate server tries to send the certificate to the node.	The certificate server tries to send the certificate to an unreachable managed node every minute after it is granted. After the time limit of two hours, the certificate request is removed from the queue. This means a new certificate request needs to be created on the managed node with `ovcert -certreq`. After a while, the certificate is successfully installed and the state changes to YES.

Managing Multiple Versions of HPOM Configuration on Managed Nodes

HPOM configuration elements, such as policies, policy groups, and instrumentation are deployed from HP Operations management server and used for monitoring applications on managed nodes.

Multiple versions of these elements can be present on different managed nodes at the same time. Monitored applications can also have multiple versions on different systems.

This section describes how you can manage multiple versions of HPOM configuration from one HP Operations management server. You can also learn how to:

- Manage the environments with predefined sets of configuration elements (for example, SPIs).
- Synchronize configuration data between different management servers.
- Manage some typical situations, described in the "Handling Multiple Versions of HPOM Configuration: Use Cases" on page 150.

For more information about policy versions, see "Policy Versions" on page 79 and the *HPOM Concepts Guide*.

For the description of the policy assignment modes and policy-assignment conflicts, see "Policy Assignment Tasks in HPOM" on page 81. For more information, see also the *HPOM Concepts Guide*.

Managing Policy Groups Versions

Policy groups cannot have version numbers assigned in HPOM, however, the version numbers can be added through the naming conventions.

You can use the following approaches:

- Adding a version number as a subgroup. For example, `/my_app/v1`, `/my_app/v2`, `/my_app/v2.1`.
- Adding the version number in the top-level policy group name (for example, `/my_app_v1`, `/my_app_v2`, `/my_app_v2.1`).

Note: Policies and the policy groups to which these policies are assigned to should have the same major number in their versions.

Managing Instrumentation Data Versions

Instrumentation files (scripts and binaries) cannot have version numbers assigned in HPOM. They should be organized by using categories. For more information, see "Category-based Distribution of Instrumentation" on page 215. For technical details, see also the *opcinstrumcfg(1m)* manual page.

If necessary, you can associate instrumentation files with numbers by using the naming conventions for categories (for example, a number in a name of a category can correspond to the application version number). However, if you build in the necessary backward compatibility into the scripts and binaries, you can reuse an existing set of instrumentation for the next version of an application, an operating system, or a custom-defined monitoring package.

You can put certain scripts or binaries into specific folders within the platform tree below each category. Note that the files from the folders that are deeper in the platform tree hierarchy overwrite the files with the same name in the folders with less specific data. For example, a file on the OS version level overwrites the file with the same name on the OS family level.

You can add the same instrumentation data with the different name also.

Note: Unused policies can still generate messages, which may cause various inconveniences. This is not a case with unused instrumentation data versions because they just occupy the disk space on the nodes.

SPI Considerations

All SPIs deliver policy versions of the format *<major_number>*.0. This means that a new SPI version always has a higher major number than the previous SPI version. Minor numbers higher than zero (>0) are reserved for customizations.

An SPI assigns all its policies to a few top-level policy groups, most SPIs use just one group. The policies are assigned to policy groups by using the `FIX` mode to avoid unexpected side effects, such as these which may occur when using the `LATEST` mode. For the information on the potential pitfalls with the `LATEST` mode, see "Potential Pitfalls with Assignment Modes" on page 82.

In addition to assigning SPI policies by using the `FIX` mode, you can also update these assignments to the `MINOR_TO_LATEST` mode. To do so, run the following command on the SPI policy groups:

```
/opt/OV/bin/OpC/utils/opcpolicy -chg_assign_mode pol_group=<SPI_group> mode=MINOR_
TO_LATEST mass_upd=yes
```

An SPI should be installed on each HP Operations management server, from which it is deployed to managed nodes. All SPI customizations can be installed later on and uploaded by using the `opccfgupld -replace -subentity` command. After uploading the SPI customizations you can, for example, move the assignment mode back to `FIX` and also move the assignments to the *.0 versions.

Flexible Management Environment Considerations

This section contains some scenarios of using and assigning policy versions in the flexible management environment.

Overwriting Policies During the Upload

The `opccfgdwn` and the `opccfgupld` utilities are used for downloading the configuration and uploading it on another server.

Policies uploaded with `opccfgupld -replace [-subentity]` always overwrite policies in the database with the same name, type, and version. In such cases a warning is printed to the `/var/opt/OV/log/System.txt` file, and also a copy of the “old” policy is stored in a subdirectory created in `/var/opt/OV/share/tmp/OpC/mgmt_sv/policy_upload_conflicts`.

Assume that a policy with version 1.0 exists on two management servers and a version 1.1 with a different content is created on both servers. Then, an upload is performed from one server to another and a policy of version 1.1 from one server overwrites the policy with the same version on another server.

To avoid overwriting policies in the flexible management environment, develop the set of policies just on one server.

Moving Policy Assignments to a Higher Version

In some cases it is necessary to move the policy assignments to a higher version after performing an upload, as shown in the following examples.

Increasing the versions of policies with direct assignments to nodes and node groups

During the upload a set of new policy versions is transferred to the target server. The policies on the target server are assigned directly through the node or node group assignments, and not through policy groups.

These nodes and node groups are not known on the development server from which the policies are uploaded. A policy with version 1.0 is assigned directly to a node on the production server by using the `FIX` assignment mode. During the upload, a policy version is increased to 1.1.

Increasing the versions of policies from an SPI customization package

An SPI customization package consists only of policies, there are no policy groups. Some customizations are uploaded to the target server. A policy with version 1.0 is assigned to the group `/my_SPI` by using the `FIX` assignment mode. A policy version 1.1 is created on the development server and uploaded.

In both examples the assignments are not a part of the upload, and the assignment mode used on the server is `FIX`.

In case the policy versions that was present on the target server before upload have the same major number as the new policy, run the following command:

```
/opt/OV/bin/OpC/utils/opcpolicy -upd_pol_assigns from=ALL to=MINOR_TO_LATEST
```

This option updates all `FIX` mode assignments to the highest minor number and keeps the major number unchanged. It does not change the assignment mode to `MINOR_TO_LATEST`. For technical details, see the *opcpolicy* manual page.

Handling Multiple Versions of HPOM Configuration: Use Cases

This section contains some use cases related to the multiple versions of the HPOM configuration on managed nodes. From the following sections you can learn how to address issues related to these cases.

Downloading and Uploading Policy Groups that Contain Policies

You can download policy groups that contain policies with their associated instrumentation categories to a file system (for example, you can upload them on a different server).

For example, to download the subgroup v3 from `/my_app`, type the following:

```
/opt/OV/bin/OpC/utils/opcpolicy -download pol_group=/my_app/v3 dir=<download_dir>
```

You can copy the data to another server and upload it by typing the following:

```
/opt/OV/bin/OpC/utils/opcpolicy -upload dir=<upload_dir>
```

If the `/my_app` top-level group does not exist, it is created, and the `/my_app/v3` subgroup is created below. The existing subgroups (for example, `/y_app/v2`) are not changed, also if you use the `opcpolicy` option `mode=replace`.

If the `/my_app/v3` subgroup already exists, it is merged with the newly created `/my_app/v3` subgroup. If the `opcpolicy` option `mode=replace` is used the elements that already exist are replaced, and missing elements are added. In any case, no elements are overwritten or deleted.

You can perform a cleanup by using the `opccfgdwn` and the `opccfgupld -replace` (without `-subentity`) utilities. For more details, see the *opccfgdwn(1m)* and *opccfgupld(1m)* manual pages.

Note: The `opcpolicy` utility can be used to transfer some configuration between servers, but it cannot completely replace the synchronization of configuration elements provided with the `opccfgdwn` and the `opccfgupld` utilities.

Listing and Removing Policies and Policy Groups

To obtain a report of all existing policy assignments, type the following:

```
/opt/OV/bin/OpC/utils/opcpolicy -list_pol_assigns
```

Note: If you want the policy type name to be listed, use the `-list_pol_assigns` option with the `level` attribute. In this case, you must set it to `2` (that is, `level=2`). For example:

```
opcpolicy -list_pol_assigns group=TEST level=2
--------------------------
assigned policy: TestPolicy, version 0001.0000, type Scheduled_Task, FIX to
policy group: /TEST
```

If you set the `level` attribute to `1` or do not set it at all, the policy type name is not listed.

You can also get the report for the specific policies, policy groups, nodes, and node groups. For example, to check where `opcmsg(1|3)` is used, type the following:

```
/opt/OV/bin/OpC/utils/opcpolicy -list_pol_assigns pol_type=msgi pol_name="opcmsg
(1|3)"
```

To list the unused (not assigned) policies, type the following:

```
/opt/OV/bin/OpC/utils/opcpolicy -list_unassigned
```

To list all policies or all versions of a given policy, type the following:

```
/opt/OV/bin/OpC/utils/opcpolicy -list_pols [ pol_type=<type> pol_name=<name> ]
```

Removing All Policies in a Policy Group

If there is a need to remove all policies assigned to some policy group, follow the procedure:

1. Get all assignments by typing the following:

   ```
   /opt/OV/bin/OpC/utils/opcpolicy -list_pol_assigns pol_group=<policy_group_name>
   ```

2. Check whether the listed policies have assignments to other policy groups. Type the following:

   ```
   /opt/OV/bin/OpC/utils/opcpolicy -list_pol_assigns pol_type=<type> pol_
   name=<name> version=<version>
   ```

3. Remove the policies by typing the following:

   ```
   /opt/OV/bin/OpC/utils/opcpolicy -remove pol_type=<type> pol_name=<name>
   version=<version>
   ```

   If necessary, delete also the previous minor versions (that have the same major number) of these policies by using the same command.

4. Remove also the policy group to which the removed policies were assigned to. Type the following:

   ```
   /opt/OV/bin/OpC/utils/opcpolicy -del_group group=<policy_group_name>
   ```

For information about removing policies in backup server flexible management environments (`-delalltempls` option of `opccfgupld`), see the *opccfgupld(1m)* manual page.

Viewing Policy Assignment Conflicts Between Server and Nodes

Sometimes the policy versions assigned on a server to be updated and deployed to a node do not match the actual policy versions on this node after deployment. The reasons for this can be various: human mistakes, configuration update failure, or conflicted policy assignments.

You can view the policy assignment conflicts by using `opcpolicy -list_conflicts`. For usage details, see the *opcpolicy* manual page.

To view the policies that should be deployed to a node, type the following:

```
/opt/OV/bin/OpC/utils/opcpolicy -list_resolved_assigns node_name=<name>
```

Match the assignments on a server against the current status on the node by typing the following:

```
/opt/OV/bin/ovpolicy -list -host <name>
```

Massive Updates of the Policy Assignment Mode

To perform massive updates of the policy assignment mode to nodes, node groups and policy groups, type the following:

```
opcpolicy -chg_assign_mode <selection> mass_upd=yes
```

Where *<selection>* is an object which the assignment refers to (for example, nodes, node groups, policy groups and so on).

For usage details, see the *opcpolicy* manual page.

Caution: Do not set the LATEST mode in production environments by using this command.

The LATEST mode is recommended for test environments only, because using different versions of configuration packages and applications would cause such assignments to be automatically switched to the highest version of a policy.

The MINOR_TO_LATEST mode can be used in production environments, but with caution. For more information about policy assignment modes, see "Policy Assignment Tasks in HPOM" on page 81. For more information, see also the *HPOM Concepts Guide*.

Massive Updates of FIX Mode Assignments

All SPIs have FIX mode assignments to *.0 versions. When customizations are done, new minor numbers of the SPI *<major>*.0 policies are created. These policies are not automatically deployed to a node, because the assignment still points to the *.0 version.

To enable deployment of these customizations, you need to move the assignment to a higher policy version. The same applies for custom policy packages with FIX mode assignments specified.

To list what would be updated according to the given input, use the following command:

```
/opt/OV/bin/OpC/utils/opcpolicy -list_pol_assigns [ <selection> ] [ filter=<from> ]
[ to=<to> ]
```

With the following command you can perform massive updates to move the assignments to the appropriate policy versions:

```
/opt/OV/bin/OpC/utils/opcpolicy -upd_pol_assigns [ <selection> ] from=<from>
to=<to>
```

For usage details and examples, see the *opcpolicy* manual page.

Moving and Renaming Policy Groups

You can rename a policy group as follows:

1. Copy the policy group by using the opcpolicy command with the -copy_group option. For usage details, see the *opcpolicy* manual page.
2. Remove the original policy group.

When policy groups are renamed, all assignments related to them are kept, such as assignments to nodes, to node groups, and to other policy groups.

To move a policy group, for example, from /my_app to /my_app/v1.0, type the following:

```
/opt/OV/bin/OpC/utils/opcpolicy -copy_group pol_group=/my_app to_group=/tmp_grp

/opt/OV/bin/OpC/utils/opcpolicy -del_group pol_group=/my_app

/opt/OV/bin/OpC/utils/opcpolicy -copy_group pol_group=/tmp_grp to_group=/my_
app/v1.0
```

Listing the Modified SPI Policies

Modified SPI policies are policies with a minor version number >0. To list them, type the following:

```
/opt/OV/bin/OpC/utils/opcpolicy -list_group pol_group=<policy_group_name>
```

To list all unchanged policies (for example, for major number 4 of an SPI), type the following:

```
/opt/OV/bin/OpC/utils/opcpolicy -list_pol_assigns pol_group=<related_group>
filter=4.0
```

Multiple Versions in Testing and Development Environments

Consider the following scenarios:

- Developing test versions in production environments

  If you do not have a test environment, and you develop your test versions of policies on production systems, use different major policy version numbers for the test policies.

  For example, if you have a production policy with version 1.3, do not give the version number 1.4 to a test policy, because the assignments (both `LATEST` and `MINOR_TO_LATEST`) would automatically move from the version 1.3 to 1.4. This results in bringing the test policy to the nodes and causes unexpected side effects.

  > **Note:** The stable test version can be moved to a preferred version later, by using the `opcpolicy -copy_pol` command.

- Policy versions in a development environment

  If a development environment is available, you can assign custom policies using the `LATEST` mode. In this case, the assignments are moved to a higher version automatically.

  After the development is finished, all final policies should have one common version number, and you may rename them according to the common naming scheme.
  You can then assign these versions to a policy group, and store this policy group (with policies and instrumentation) as a package that can be downloaded to other management servers.

- SPI policies in a development environment

  If SPI policies are evaluated in a development environment, set their assignment mode to `MINOR_TO_LATEST` (or leave it on `FIX`). For example, type the following:

  ```
  /opt/OV/bin/OpC/utils/opcpolicy -chg_assign_mode pol_group=<SPI_group>
  mode=MINOR_TO_LATEST mass_upd=yes
  ```

After you perform the customizations, remove temporary policy versions and assign an appropriate number to the final customized version. For example, when the SPI policy version 4.0 is customized for the first time, the version number changes to 4.1. Later updates for production purposes may increase this number to 4.2.

Conveying Policies and Policy Groups from Development to Production

Assume the `p_basic` and the `p_advanced` policies are created for the `my_app` application version 4.0. Version 4.5 of `my_app` is introduced, which contains some major changes.

Some servers still run the `my_app` application version 4.0, some already run the version 4.5. Assume that the `p_basic` policy can monitor the application version 4.5, and the `p_advanced` policy requires some updates. Both policies may be enhanced over time and policy patches must be installed on the production servers. In addition some customizations are necessary for specific systems (for example, thresholds need to be adapted).

- *On the HPOM development server*
  a. Create the `/my_app/v4` and `my_app/v4.5` policy groups.
  b. Give the major number 400 to the `p_basic` and the `p_advanced` policies and assign both policies to the `/my_app/v4` policy group.
  c. Update the `p_advanced` policy to be able to monitor the application version 4.5, and give it a major version 450.
  d. For the sake of simplicity, duplicate the `p_basic` policy from 400.0 to 450.0, so that all policies in the `/my_app/v4.5` group have the same major version.
  e. Applying a policy patch increases the minor version, so assume that the versions 450.1 and 450.2 of the `p_advanced` policy and the version 450.1 of the `p_basic` policy are created over time. Assign them only to the `/my_app/v4.5` policy group and not to `/my_app/v4`.

     Tip: You could assign these policies with the `MINOR_TO_LATEST` mode, so that a new minor version is automatically selected for deployment as soon it is uploaded to the management server.

     Create or update an upload package that includes everything needed to monitor the application (in this scenario, `my_app`). The upload package can contain appropriate policy groups, instrumentation categories, application definitions, and so on.
- *On the HPOM production server*
  a. Upload the configuration package by using the `opcpolicy -replace -subentity` command.
  b. Assign the `/my_app/v4` policy group to the nodes that run the `my_app` application version 4. Similarly, assign the `/my_app/v4.5` policy group to the nodes that run the `my_app` application version 4.5.

c. Distribute the configuration to all managed nodes.

> **Tip:** If you need to make customizations for some monitored systems (for example, to change thresholds), it is not recommended to use new version numbers of the `p_basic` and the `p_advanced` policies for this purpose.
>
> Instead, you can create `nodeinfo` policies with appropriate name-value pairs, which overwrites the default settings in the `p_basic` and the `p_advanced` policies.
>
> You can assign these `nodeinfo` policies directly to nodes or node groups, it is not necessary for them to be part of the upload package. Their names may correspond to the names of policies for which they are created (for example, `p_advanced_settings_for_critical_systems`).

Rollback to Previous Versions

Performing a Rollback to a *.0 Version

You can perform rollback to a *.0 version of an SPI or a custom configuration package by setting back the assignment mode to `FIX` (if the *.0 version was also assigned with `FIX`) and then moving the assignments.

For example, assume a policy group `/my_app/v4` has all its policies assigned with `MINOR_TO_LATEST` mode, and the major number is 400.

To perform a rollback to a *.0 version, type the following:

```
/opt/OV/bin/OpC/utils/opcpolicy -chg_assign_mode pol_group=/my_app/v4 mode=FIX
/opt/OV/bin/OpC/utils/opcpolicy -upd_pol_assigns pol_group=/my_app/v4 from=400.ALL to=400.0
```

Performing a Rollback to an Intermediate State

For the policies without *.0 versions the rollback has to be performed to an intermediate state, as follows:

1. Switch to the `FIX` mode as follows:

   ```
   /opt/OV/bin/OpC/utils/opcpolicy -chg_assign_mode pol_group=/my_app/v4 mode=FIX
   ```

2. Change the assignment mode to `FIX` inside the package for upload as follows:
   a. Change directory to `POLGROUPS`:

      ```
      cd <upload_package_dir>/<language>/POLGROUPS
      ```

   b. In the files with names starting with "PolicyConfig" specify the value `fixed` for the

<assignment_mode> XML tags.

You can remove the higher policy versions for the same major numbers after the upload.

3. To upload the package with the intermediate state, type the following:

```
opccfgupld -replace -subentity
```

Caution: The uploaded package must not contain `MINOR_TO_LATEST` assignments. They would lead back to the higher version numbers because the corresponding policies are still installed on the management server.

Performing a Rollback to a Different Major Version

Assume policy groups `/my_app/v4` and `/my_app/v4.5` have all policies assigned with the `MINOR_TO_LATEST` mode.

To perform a rollback from version 4.5 to 4, deassign the `/my_app/v4.5` policy group from the respective nodes or node groups, and assign the `/my_app/v4` policy group.

When you remove one policy that causes problems (for example, version 450.1), assuming that it is assigned with the `MINOR_TO_LATEST` assignment mode, an automatic rollback to a previous version (in this example, version 450.0) is performed.

Note: When the last version of a major number is removed, the `MINOR_TO_LATEST` assignment is also removed. It is also removed when you switch to the `FIX` mode and remove the version.

Working with HTTPS Managed Nodes

This section describes the following tasks related to HTTPS managed nodes:

- "Defining Common Settings for Managed Nodes" on page 157
- "Allocate a Specific OvCoreId to a Managed Node" on page 158
- "Configuring a Windows Installation Server" on page 158

Defining Common Settings for Managed Nodes

You can define settings on the management server, which are deployed to the managed nodes at installation time. Basic parameters, such as communication ports or HTTP proxy settings that are used by many nodes can be defined this way. Common scenarios include:

- Need to install many HPOM agents on a subnet or domain. Due to firewall restrictions, the default port of the Communication Broker (383) cannot be used and you want to avoid having to manually set the Communication Broker port on every node during agent installation.

- Configure default settings for installation of managed nodes at a central point as the nodes of a subnet or domain share many settings.
- HPOM agents are manually installed on a subnet behind a firewall. Common parts of the installation can be automated.

You can maintain these common settings on the HP Operations management server using the file:

`/etc/opt/OV/share/conf/OpC/mgmt_sv/bbc_inst_defaults`

A sample configuration file with examples of how to set up parameters is available at:

`/etc/opt/OV/share/conf/OpC/mgmt_sv/bbc_inst_defaults.sampl`

Take a copy of `bbc_inst_defaults.sampl`, rename it `bbc_inst_defaults`, and modify in accordance with the syntax specified in the sample file.

Allocate a Specific OvCoreId to a Managed Node

If you want to allocate a specific OvCoreId for a new node, manually add it as follows before starting the agent software installation:

On the HP Operations management server, enter one of the following commands:

- `opcnode -chg_id ... id=<id>`
- `opcnode -add-node ... id=<id>`

During agent installation, the OvCoreId from the HPOM database is used for the specified managed node.

This is recommended when reinstalling a node managed by many management servers. Reusing the original OvCoreId avoids having to update all the HP Operations management servers.

When installing certificates manually, everything is prepared on the HP Operations management server before an agent is installed, including creating an OvCoreId, generate a certificate, add the node with the new OvCoreId to the database. Only after these steps can the agent software be installed on the managed node. Finally the certificate must be copied to the managed node.

Configuring a Windows Installation Server

HTTPS agents can be fully automatically installed onto Windows systems using an installation server system. An installation server is a regular Windows managed node with an HTTPS agent installed. Once the HTTPS agent is installed, you can install any further Windows HTTPS nodes using `inst.sh` on the HP Operations management server without the need to manually execute the `oasetup.exe` utility on the target nodes.

Note: It is necessary to set the installation server of the target nodes.

The following guidelines describe the specific configurations required for the HTTPS agent acting as installation server:

- The Windows system hosting the HTTPS agent which acts as installation server must be in the HPOM node bank and must be of the same communication type (HTTPS) as the target nodes.
- It is recommended to use a dedicated system as an installation server system because it is necessary that the HTTPS agent acting as the installation server runs with extensive capabilities (see below). This means that this agent should not receive any policies or instrumentation to avoid accidental or malicious start of functionality with these capabilities.
- The HTTPS agent must run as a user who is able to access the target systems using standard Windows access mechanisms. In particularly it must be able to copy files to the target system as the software is transferred to the Windows nodes using a windows share.

To configure a managed node to act as the Windows Installation Server, complete the following steps:

1. Install and start the Windows service on the target system. This can be accomplished by making this agent run as either:
   - A domain administrator
   - Any other user who has:
     - Networking capabilities.
     - Windows pass-through authentication is in place (identical user/password on both nodes).
     - Administrative capabilities on the target nodes.

   Tip: For information about Windows user rights and privileges, see the Microsoft documentation at the following location:

   `http://www.microsoft.com/technet/security/prodtech/`

   To install Windows agent software using an installation server, the HTTPS agent acting as the installation server cannot run as SYSTEM (which is the default) because it is not able to access remote systems. Instead, this agent must run under an identity, which is able to access the target managed node using regular Windows access mechanisms to the admin drive.
2. Change the user under which the HTTPS agent acting as an installation server runs:
   a. Stop the HTTPS agent with the command:

      ```
      ovc -kill
      ```
   b. Create the Windows user account to be used.
   c. Make the following user and permission changes to the selected Windows user account to make sure that the agent is running with the appropriate privileges as well as the agent directory structure has the appropriate privileges set:

- Changes the start-up user of the Windows Service.
- Change the permissions of HPOM data files.

  Enter the following command:

  ```
  cscript <InstallDir>\bin\ovswitchuser.vbs -existinguser <user> -
  existinggroup <group> -passwd <user_pwd>
  ```

  This command requires a few minutes to execute.

d. Due to a limitation in `ovswitchuser.vbs`, complete the following steps:

   i. Open `Control Panel` -> `Administrative Tools` -> `Services`.

   ii. Change the Windows user to one which is configured to run the service `HP OpenView Ctrl Service` and re-enter the user password.

   > **Note:** The `SYSTEM` account is not sufficient to do the install-server tasks as it does not have the appropriate network rights. Because of this, you must change the agent user on the installation server to an existing administrative account with sufficient network rights. This user is not created automatically.

   iii. Confirm that the user has been given the `Start as service` capability.

e. Start the agent with the command:

   ```
   ovc -start
   ```

f. Verify that the processes are running and note the user under which they are running as follows:

   i. `ovc -status`

   ii. Open the `Task Manager` and display the user.

Working with Virtual Nodes

The following sections describe how to work with virtual nodes in HPOM:

- "Getting the First Message for a Virtual Node" on page 163
- "Monitoring HARGs in the Java UI" on page 167

Adding Virtual Nodes to HPOM

Virtual nodes can be configured in a node bank by uploading them with the `opccfgupld(1m)` utility or the `opcnode(1m)` utility.

The new call parameters added to `opcnode(1m)`:

```
-set_virtual
node_list = "node1 node2 …"
cluster_package = HARG_name
```

Example:

```
./opcnode -set_virtual node_name=ovguest3 node_list="talence ovguest3" cluster_
package=HARG_name
```

Note: All nodes that are to be a part of a cluster must also be members of the node bank. They must all share the same node type characteristics (platform, operating system, communication type). The virtual node must not be a DHCP node.The physical nodes of a cluster must not be virtual nodes themselves.

Modifying Virtual Nodes in HPOM

To modify the virtual node-related information, enter the following commands:

- To change the HA Resource Group name:

  ```
  opcnode -set_virtual node_name=<virtual host> cluster_package=<HA resource group>
  node_list=<physical nodes>
  ```

- To change the list of physical nodes:

  ```
  opcnode -set_virtual node_name=<virtual host> node_list=<physical nodes>
  ```

Note: All nodes that are to be a part of a cluster must also be members of the node bank. They must share the same node type characteristics (platform, operating system, communication type).

The physical nodes of a cluster must not be virtual nodes themselves.

Assigning Policies to Virtual Nodes in HPOM

Assigning policies to virtual nodes is done in the same way as assigning policies to physical nodes, that is, by using the `opcnode` command line utility. For example:

```
opcnode -assign_pol node_name=<virtual_node> net_type=NETWORK_IP pol_name=<policy_
name> pol_type=<policy_type> [ version=<version> ]
```

For more information about the policy assignment, see the *HPOM Concepts Guide* and the *opcnode (1m)* manual page.

Deploying Policies to Virtual Nodes in HPOM

To deploy policies to virtual nodes, use the `opcragt` command line utility. For example:

```
opcragt -dist <virtual_node>
```

Note: The HPOM agent software cannot be deployed to a virtual node. It must be installed on all physical nodes, which make up the virtual node.

For more information about the policy deployment, see the *HPOM Concepts Guide* and the *opcragt* manual page.

Modifying Policy Configuration on Virtual Nodes in HPOM

To modify a policy, follow these steps:

1. Download the policy by using the `opctempl` tool with the `-download` command line argument.

   For more information, see the *opctempl(1m)* manual page.
2. Edit the policy body by using your favorite editor.
3. Make modifications according to the policy body grammar.

   For detailed information about the policy body grammar for the default policy types, see the *HPOM Concepts Guide*.
4. After you have made modifications, save the policy body and upload the policy by using `opctempl -upload`.

Note: Make sure that you do not make changes to the policy header, otherwise the upload might fail. When the policy is uploaded, a new version with the modified policy body is created.

Deassigning Policies from Virtual Nodes in HPOM

To deassign policies from virtual nodes, use the `opcnode` command line utility. For example:

```
opcnode -deassign_pol node_name=<virtual_node> net_type=NETWORK_IP pol_
name=<policy_name> pol_type=<policy_type> [ version=<version> ]
```

Deleting Virtual Nodes from HPOM

To delete a virtual node from the node bank, use the `opcnode` command line utility. For example:

```
opcnode -del_node node_name=<virtual_node> net_type=<network_type>
```

Configuring Agents on Multi-homed Hosts

For some physical nodes, for example for multihomed nodes, the standard hostname may be different from the name of the node in the cluster configuration. If this is the case, the agent cannot correctly determine the current state of the resource group.

Configure the agent to use the hostname as it is known in the cluster configuration:

1. On the physical node, run the `ovclusterinfo -a` command to obtain the name of the physical node as it is known in the cluster configuration.
2. Configure the agent to use the name of the node as it is known in the cluster configuration:

   ```
   ovconfchg -ns conf.cluster -set CLUSTER_LOCAL_NODENAME <name>
   ```
3. Replace `<name>` with the name of the node as reported in the output of `ovclusterinfo -a`.
4. Stop the agent:

   ```
   ovc -stop AGENT
   ovc -stop COREXT
   ```
5. Start the agent:

   ```
   ovc -start COREXT
   ovc -start AGENT
   ```

Getting the First Message for a Virtual Node

This is an example to generate a message for a virtual node. Prerequisite is an HA cluster on which one or more HA resource groups are running. For simplicity, just select one of the existing resource groups and model it in HPOM as a virtual node. You need to know the resource group name, either the IP address or nodename to do this.

1. Make sure that the HPOM agent software is installed on each physical node of the cluster.
2. Add the virtual node into the node bank.
3. Add the physical nodes belonging to the virtual node.
4. Specify the HA resource group name associated with the virtual node.

   In the following steps, the HA resource group name is referred to as `<my_resource_group>`.
5. Configure CMAs in the policy.

a. Download the policy by using the `opctempl` tool with the `-download` command line argument.

   For more information, see the *opctempl(1m)* manual page.

b. Edit the policy body by using your favorite editor.

c. Make modifications according to the policy body grammar.

   For detailed information about the policy body grammar for the default policy types, see the *HPOM Concepts Guide*.

d. After you have made modifications, save the policy body and upload the policy by using `opctempl -upload`.

   Note: Make sure that you do not make changes to the policy header, otherwise the upload might fail. When the policy is uploaded, a new version with the modified policy body is created.

6. Assign the `opcmsg(1|3)` policy to the virtual node.
7. Distribute the `opcmsg(1|3)` policy to the virtual node.
8. Check if the policy is installed on the agent using the `ovpolicy` command.

   On each physical node, enter the command:

   ```
   ovpolicy -l -level 4
   ```

   For example, the following information is displayed:

   ```
   msgi      "opcmsg(1|3)"   <enabled or disabled> 0009.0000
   policy id : "15012f6e-ab2a-71d9-1d2e-0a110b850000"
   owner     : "OVO:<full_qualified_virtual_node_name>"
   category  : <no categories defined>
   attribute : "HARG:<my_resource_grp_name>" "no_value"
   ```

   Note: If the policy is assigned to the virtual node only, on the node where the HA package is running, this policy is enabled. On the node where the HA package is not running, this policy is disabled.

   You can obtain policy status information (enabled or disabled) using the command `ovpolicy -l`.

   For example, to list installed policies for the local agent, enter the command:

   ```
   ovpolicy -l
   ```

   In this case, the information is displayed in the following form:

   ```
   * List installed policies for host 'localhost'.

   Type             Name             Status     Version
   ------------------------------------------------------
   configsettings   "OVO settings"   enabled    1
   ```

```
msgi            "opcmsg(1|3)"    enabled    0009.0000
monitor         "mondbfile"      disabled   0009.0000
```

9. Check whether the `apminfo.xml` file is already installed on each physical node.

   On the management server, execute the following command for each of your physical nodes:

   ```
   "
   for node in <all your physical nodes>
   do
   opcdeploy -cmd "ls"  -par "\$OvConfDir/conf/apminfo.xml" -node $node
   done
   "
   ```

10. If the `apminfo.xml` file is NOT installed, edit the `apminfo.xml` file on the management server and install it on each physical node as follows:

    a. `cd /tmp`

    b. `vi apminfo.xml`

    c. Put the following contents into the `apminfo.xml` file and save the file:

    ```
    "
    <?xml version="1.0"?>
    <APMClusterConfiguration>
          <Application>
                <Name>OpenView_Application</Name>
                <Instance>
                      <Name>openview</Name>
                      <Package>ov-server</Package>
                </Instance>
          </Application>
    </APMClusterConfiguration>
    "
    ```

    Note: The extract for the `apminfo.xml` file mentioned above is an example, and the application `OpenView_Application` is defined, which is mapped to `my_ns` defined in a CMA. It also defines the mapping between the application instance `openview` and the HA Resource Group `ov-server`. Instance `openview` is mapped to `my_instance` defined in the CMA.

    d. Install the `apminfo.xml` file on each physical node as follows:

    ```
    "
    for node in <all of your physical node names>
    do
    opcdeploy -deploy -file /tmp/apminfo.xml -node $node -targetdir "conf/conf"
    -trd data
    done
    "
    ```

11. If the `apminfo.xml` file is already installed on the agent, you must edit the existing `apminfo.xml` file manually as follows:
    a. Log on to the system where the `apminfo.xml` file is installed.
    b. `cd \$OvConfDir/conf/`
    c. `vi apminfo.xml`
    d. Keep the existing application definitions and define your application:

```
"
<?xml version="1.0"?>
<APMClusterConfiguration>
    <Application>
        <Name>Existing_Application</Name>
        <Instance>
            <Name>Existing_instance</Name>
            <Package>Existing_resource_group_name
                                        </Package>
        </Instance>
    </Application>
    <Application>
        <Name>OpenView_Application</Name>
        <Instance>
            <Name>openview</Name>
            <Package>ov-server</Package>
        </Instance>
    </Application>
</APMClusterConfiguration>
"
```

12. Configure the file:

    `$OvDataDir/bin/instrumentation/conf/<appl_name>.apm.xml`

    This file should be created in the directory:

    `$OvDataDir/bin/instrumentation/conf`

    on each physical node and it should take the form of the following example:

```
<?xml version="1.0"?>
<APMApplicationConfiguration>
    <Application>
        <Name>OpenView_Application</Name>
        <Template>opcmsg(1|3)</Template>
    </Application>
</APMApplicationConfiguration>
```

    For more detailed information about configuring the `<appl_name>.apm.xml` file, see the HP Operations agent documentation.

13. If the `opcmsg(1|3)` policy is installed on the agent and `enabled`, and if the `apminfo.xml` file is

installed, execute the following command from this agent:

```
opcmsg a=a o=testcma msg_t="I want to test CMA" -option my_ns=OpenView_
Application -option my_instance=openview
```

You should receive a normal message for the virtual node with the following details in the browser:

```
Node:<virtual_nodename>
Application: "a"
Object: "testcma_result"
Message Text: "Receive enriched message from CMA"
```

Monitoring HARGs in the Java UI

Clusters and their nodes can be monitored in the `Services Graph` window. You can configure the cluster so that the active node is labelled with, for example, the application that it is hosting. When this node is no longer active, the label is switched to the new active node.

To monitor HA resource groups in the Java UI, the following configurations need to be made:

- Create an APM definition file to define the mappings between HA resource groups and application instances.
- Create or configure a command, script or executable, which is run when an HA resource group is started or stopped.
- Specify start and stop hooks used by APM and CLAw to execute additional tasks at HA package switch or fail over.
- Configure the custom message attributes.
- Create policies to label and unlabel the system in the Java GUI on which the HA resource group is active or inactive when an HA resource group is started or stopped.

Our example is based on a cluster `tommy2`, consisting of two physical nodes, `tcbbn092` and `tcbbn093`. Three HARGs are installed on this cluster; `OpenView_Application`, `second-rg` and `third-rg`. This example concentrates on the `third-rg` application. To be able to monitor HARGs in the Java GUI, the following steps need to be provided.

1. Create the APM definition file to define the mappings between HA resource groups and application instances. In the following example, for simplicity, we configure the application name and instance name to be the same as the HARG name for HA resource group "`second_rg`" and "`third_rg`". For further information, see the HP Operations agent documentation.

   ```
   # more /var/opt/OV/conf/conf/apminfo.xml

   <?xml version="1.0"?>
   <APMClusterConfiguration>
         <Application>
               <Name>OpenView_Application</Name>
               <Instance>
   ```

```
                <Name>openview1</Name>
                <Package>ov-server</Package>
          </Instance>
    </Application>
    <Application>
          <Name>second-rg</Name>
          <Instance>
                <Name>second-rg</Name>
                <Package>second-rg</Package>
          </Instance>
    </Application>
    <Application>
          <Name>third-rg</Name>
          <Instance>
                <Name>third-rg</Name>
                <Package>third-rg</Package>
          </Instance>
    </Application>
</APMClusterConfiguration>
```

2. Create a shell script which will be executed when a HARG is started or stopped. It will log the start and stop information to the log file `/tmp/clawapplication_log` and send a status message to the browser. The shell script should look like the following example:

```
# more /tmp/test_clawst.sh

application=$1
label=$2
start_stop=$3
echo "app=$application st=$start_stop label=$label" >>/tmp/clawapplication_log
echo "$application $start_stop at:" >>/tmp/clawapplication_log
date >>/tmp/clawapplication_log
echo "HPOM_instance is $application" >>/tmp/clawapplication_log
echo "Sending $start_stop message..." >>/tmp/clawapplication_log
/opt/OV/bin/OpC/opcmsg a=a o=o msg_t="$application $start_stop" -option
label=$label -option my_instance=$application -option my_ns=OpenView
echo "$application ends at:" >>/tmp/clawapplication_log
date >>/tmp/clawapplication_log
echo "========================" >>/tmp/clawapplication_log
```

3. Specify start and stop hooks used by APM and CLAw to execute additional tasks at HA package switch or fail over. For further information, see the HP Operations agent documentation.

   In the following example, we specify start and stop hooks for `third-rg`. When `third-rg` is started, the shell script `/tmp/test_clawst.sh` which we defined in the previous step is executed with input parameters `$instanceName ov_label3 starts`. A message with text `third-rg starts` is then sent to the browser and the value of `label` is set to `ov_label3`. When `third-rg` is stopped, the same shell script is executed with input parameters `$instanceName ov_label3`

`stops` and a message with text `third-rg stops` is then sent to the browser and the value of `label` is set to `ov_label3`.

The start and stop definitions should be specified as in the following example:

```
# more /var/opt/OV/bin/instrumentation/conf/third-rg.apm.xml

<?xml version="1.0"?>
<APMApplicationConfiguration>
   <Application>
      <Name>third-rg</Name>
      <StartCommand>
         /tmp/test_clawst.sh $instanceName ov_label3 starts
      </StartCommand>
      <StopCommand>
         /tmp/test_clawst.sh $instanceName  ov_label3 stops
      </StopCommand>
   </Application>
</APMApplicationConfiguration>
```

4. Configure custom message attributes. For more information, see the HP Operations agent documentation.
5. Create a policy to check if an HARG is started and to label the system in the Java UI on which the HARG is active. Deploy this policy to the virtual node.

   The following policy example checks the message text for a running HARG. On finding one, it runs an automatic action to label the active cluster node with the package name, `third-rg` on node `tcbbn093` in our example.

```
OPCMSG "opcmsg(1|3)

DESCRIPTION "starts HARG"
      CONDITION_ID "96a679b2-b59c-71d9-1ed2-c0a801020000"
      CONDITION
            TEXT "<*> starts<*>"
      SET
            SERVICE_NAME "<$MSG_GEN_NODE_NAME>"
            MSGKEY "<$OPTION(my_instance)>"
            MSGKEYRELATION ACK  "<$OPTION(my_instance)>"
            CUSTOM  "instance"  "<$OPTION(my_instance)>"
            CUSTOM  "namespace"  "<$OPTION(my_ns)>"
            CUSTOM  "orig_nodename"  "<$MSG_GEN_NODE_NAME>"
            AUTOACTION "/opt/OV/bin/OpC/opcsvcattr svc_id=<$MSG_GEN_NODE_NAME>
name=<$OPTION(label)> value=<$OPTION(my_instance)>" ACTIONNODE IP 0.0.0.0
"<$OPC_MGMTSV>"
ANNOTATE
            SIGNATURE "EAJHjRr9vq48…
```

   Enter the following command to run the `third-rg` HARG on the node `tcbbn093`:

```
/usr/sbin/cmrunpkg -n tcbbno93 third-rg
```

The `third-rg` HARG is started, the message `third-rg starts` is received and the icon of the node `tcbbn093` in the Java UI is labeled with the active package name, `third-rg`.

6. Create a policy to check if an HARG is stopped and to remove the label from the system in the Java UI on which the HARG was active. Deploy this policy to the virtual node.

   The following policy example checks the message text for a stopped HARG. On finding one, it runs an automatic action to remove the label from the now no longer active cluster node, `tcbbn093` in our example.

```
OPCMSG "opcmsg(1|3)

    DESCRIPTION "default interception of messages submitted by opcmsg(1) and
opcmsg(3)"
    FORWARDUNMATCHED
    MSGCONDITIONS
    DESCRIPTION "stops HARG"
        CONDITION_ID "8070b36c-b5b3-71d9-1ed2-c0a801020000"
        CONDITION
            TEXT "<*> stop<*>"
        SET
            SEVERITY Warning
            SERVICE_NAME "<$MSG_GEN_NODE_NAME>"
            MSGKEY "<$OPTION(my_instance)>"
            MSGKEYRELATION ACK  "<$OPTION(my_instance)>"
            CUSTOM  "instance"  "<$OPTION(my_instance)>"
            CUSTOM  "namespace"  "<$OPTION(my_ns)>"
            CUSTOM  "orig_nodename"  "<$MSG_GEN_NODE_NAME>"
            AUTOACTION "/opt/OV/bin/OpC/opcsvcattr -remove svc_id=<$MSG_GEN_
NODE_NAME> name=<$OPTION(label)>" ACTIONNODE IP 0.0.0.0  "<$OPC_MGMTSV>
ANNOTATE
            SIGNATURE "RgUMFg…
```

   Enter the following command to stop the `third-rg` HARG on the node `tcbbn093`:

```
/usr/sbin/cmhaltpkg -n tcbbno93 third-rg
```

   The `third-rg` HARG on the node `tcbbn093` is stopped. The message `third-rg stops` is received and the label of the package name, `third-rg`, is removed.

   On switching the `third-rg` HARG to the node `tcbbn092`, the node icon in the `Service Graph` is labeled with the application name `third-rg`.

Proxies in HPOM

Firewall programs and their associated policies, located at a network gateway server, are gateways that are used to protect the resources of a private network from external users. Users of an intranet are

usually able to access the approved parts of the Internet while the firewall controls external access to the organization's internal resources.

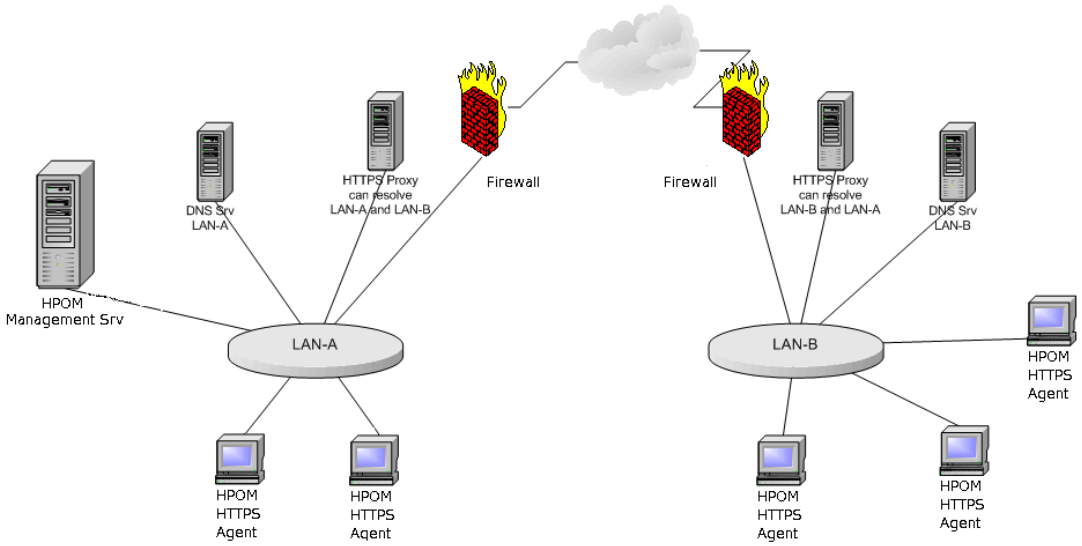
A proxy is a software application that examines the header and contents of Internet data packets and takes necessary action required to protect the systems to which the data is directed. In conjunction with security policies, proxies can remove unacceptable information or completely discard requests.

For more information about proxies in HPOM, see the *HPOM Concepts Guide*.

Configuring Proxies

Most LAN-Internet-LAN architectures can be represented by the following diagram or a subset of the illustration.

Figure 3: HTTP Proxy Schematic



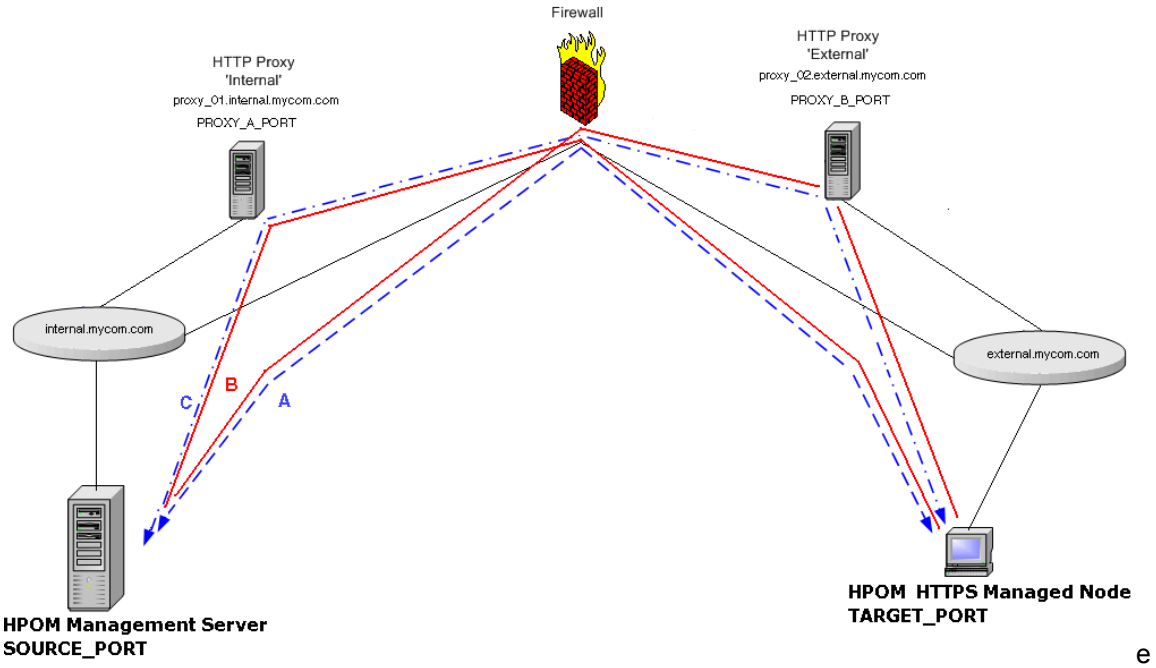
Internal LAN-A includes the HP Operations management server and an HTTP proxy.

A firewall separates the internal LAN from the Internet and the outside world.

An external LAN-B includes HTTPS managed nodes and an HTTP proxy.

The proxy communication can be represented by the following diagram or a subset of the illustration.

Figure 4: HTTP Proxy Infrastructure



e

A: Direct communication; no Proxy. Firewall must accept all connections from `*.internal.mycom.com:*` to `*.external.mycom.com:TARGET_PORT` and all connections from `*.external.mycom.com.*` to `*.internal.mycom.com:SOURCE_PORT`.

B: `proxy_01` is the proxy in domain `internal.mycom.com` and can access domain `external.mycom.com`. Firewall must accept all connections from `proxy_01.internal.mycom.com:*` to `*.external.mycom.com:TARGET_PORT`.

`proxy_02` is the proxy in domain `external.mycom.com` and can access domain `internal.mycom.com`. Firewall must accept all connections from proxy_01.internal.mycom.com to `*.internal.mycom.com:SOURCE_PORT`.

C: `proxy_01` is the proxy in domain `internal.mycom.com`. `proxy_02` is the proxy in domain `external.mycom.com`. `proxy_01` can access `proxy_02` and `proxy_02` can access `proxy_01`. Firewall must accept all connections from `proxy_01.internal.mycom.com:*` to `proxy_02.external.mycom.com:PROXY_B_PORT` and `proxy_02.external.mycom.com:*` to `proxy_01.internal.mycom.com:PROXY_A_PORT`.

The proxies through which a managed node is to communicate must be specified for each system. This is set in the namespace `bbc.http` and stored in the `bbc.ini` file using the `ovconfchg` command. `bbc.ini` must not be edited manually.

Syntax

```
ovconfchg -ns <namespace> -set <attr> <value>
```

In this instance:

`-ns <namespace>`	Sets a namespace for following options.
`-set <attr> <value>`	Sets an attribute (proxy) and values (port and addresses) in current namespace.

For example:

```
ovconfchg -ns bbc.http -set PROXY “web-proxy:8088-(*.mycom.com)+(*.a.mycom.com;*)“
```

Defines which proxy and port to use for a specified hostname.

Format: `proxy:port +(a)-(b);proxy2:port2+(a)-(b); ...;`

`a`: list of hostnames separated by a comma or a semicolon, for which this proxy shall be used.

`b`: list of hostnames separated by a comma or a semicolon, for which the proxy shall not be used.

The first matching proxy is chosen.

It is also possible to use IP addresses instead of hostnames so `15.*.*.*` or `15:*:*:*:*:*:*:*` would be valid as well, but the correct number of dots or colons MUST be specified. IP version 6 support is not currently available but will be available in the future.

```
PROXY=web-proxy:8088-(*.hp.com)+(*.a.hp.com;*)
```

The proxy `web-proxy` is used with port `8088` for every server (`*`) except hosts that match `*.hp.com`, for example `www.hp.com`. If the hostname matches `*.a.hp.com`, for example, `merlin.a.hp.com` the proxy server will be used.

Set Proxies on the HP Operations Management Server

To change the proxy settings on the HP Operations management server:

1. Set the proxies over which the management server will communicate with its HTTPS managed nodes. For example:

   ```
   ovconfchg -ns bbc.http -set PROXY “web-proxy:8088-(*.mycom.com)+
   (*.a.mycom.com;*)“
   ```

2. Stop all HP Operations processes with the following commands:

   ```
   opcsv -stop
   /opt/OV/bin/OpC/ovc -kill
   ```

3. Restart the processes with the following commands to register the proxy changes:

   ```
   opcsv -start
   /opt/OV/bin/OpC/opcagt -start
   ```

Managing HTTPS Agents on DHCP Client Systems

This section describes managing HTTPS agents on DHCP client systems, as follows:

HP Operations Agents and DHCP

Dynamic Host Configuration Protocol, or DHCP, enables a DHCP server to dynamically allocate network configurations to computers on an IP network. The primary purpose of this is to reduce the work necessary to administer a large IP network and distribute IP addresses to computers as they are required.

DHCP is a client-server application. When a computer connects to a DHCP server, the server temporarily allocates the computer an IP address. The computer uses this address until the lease expires, at which point it can be replaced with a new IP address.

The main advantage of DHCP is that its addressing scheme is fully dynamic. With a DHCP server running on your network, you can add or move computers around on your network and not have to worry about re-configuring your IP settings.

You can manage HTTPS agents running on DHCP-Client systems. The HPOM solution is not dependent on any specific DHCP or DNS product and is based on the following assumptions:

- System names must not change. The system name can be used as an identifier of a system, even in a flexible management environment.
- DHCP and DNS are synchronized.
- There are a relatively small number of IP address changes per day so no IP Address Change Event (IPCE) Storm strategy is necessary. An HP Operations agent sends this event, when it detects an IP address change on one of its network interfaces.
- The Java GUI processes do not automatically update the IP address changes.
- DHCP support of agents is configurable for each agent and server.
- Dynamic IP address changes at runtime, not only at startup.

  The time between two IP address change checks can be configured by setting the `IPADDR_CHECK_INTERVAL` variable on the system.

DHCP Settings in HPOM

Variables for DHCP

The following variables are used to configure the DHCP-specific behavior of the management server processes.

`OPC_DUMMY_IP_RANGE 1.1.1.*`

If the HPOM for UNIX management server detects an IP address conflict while processing an IP change request, the next free IP address out of the OPC_IP_DUMMY_IP_RANGE is used. The format of this string is [1-9*].[1-9*].[1-9*].[1-9*]. At least one number must be specified. The default is 1.1.1.*.

OPC_IPCE_RETRY_NUM 10

If none of the IP addresses reported by t he system matches those of DNS, the IP address change event is buffered. Each event is processed with a maximum number of retries as specified by the OPC_IPCE_RETRY_NUM variable. The default is 10.

OPC_IPCE_RETRY_INTERVAL 180

After the OPC_IPCE_RETRY_INTERVAL time period has elapsed, all buffered IP change events are processed again. The default is 180 seconds.

Using opcnode for DHCP

You can use the opcnode command to specify the DHCP. To configure the HP Operations management server to accept IP address change events, set dynamic_ip to yes as follows:

- When adding a new node:

  ```
  $ opcnode -add_node node_name=<node name> dynamic_ip=yes net_type=<net_type>
  mach_type=<mach type> group_name=<group name>
  ```

- When modifying a system:

  ```
  $ opcnode -chg_iptype dynamic_ip=yes -node_name=<node_name> | -node_list='<list>'
  ```

Note: The network type of all specified nodes must be NETWORK_IP. It is not possible to specify another network type with net_type.

Enabling Management of Agents on DHCP Clients

To enable management of HTTPS agents on DHCP Clients, ensure that DHCP and DNS are synchronized (for example, by updating from the DHCP Server). If synchronization is not achieved, the HP Operations management server cannot process any IP address change events, and it decreases the overall performance of the system.

Managing Variables in HPOM

When managing variables in HPOM, you can set, read, or delete variables, as well as customize XPL config variables locally.

You can find more documentation and examples about configuration settings in the files that can be found at the following location:

/opt/OV/misc/xpl/config/defaults/*.ini

Setting Variables

To set variables on the HP Operations management server, run the following command:

```
/opt/OV/bin/ovconfchg -ovrg server -ns opc -set <var_name> <value>
```

All relevant variables that were available in the `opcsvinfo` files are also used by HPOM.

The HPOM schema uses namespaces (the `-ns` parameter from the example above). All former `opcsvinfo` variables now have the namespace `opc`, all former `opcinfo/nodeinfo` variables on HTTPS nodes have the namespace `eaagt`.

You can suffix the namespace by the process name if required. For example, to set the maximum number of simultaneous connections to `opcuihttps`, run the following command:

```
ovconfchg -ovrg server -ns opc.opcuihttps -set MAX_CONNECTIONS 200
```

Reading Variables

To read the variables on the HP Operations management server, run the following command:

```
/opt/OV/bin/ovconfget -ovrg server [ <namespace> [ <var_name> ] ]
```

This either prints all settings, all settings of a namespace, or one variable.

To read variables on a managed node, use the `ovconfget` command without the `-ovrg server` option.

Deleting Variables

To delete the variables on the HP Operations management server, run the following command:

```
/opt/OV/bin/ovconfget -clear [ <namespace> [ <var_name> ] ]
```

Troubleshooting HTTPS Agents

This section contains useful information about troubleshooting HTTPS agents, which is organized within the following sections:

- "Troubleshooting HTTPS-based Communication" on page 177
- "Troubleshooting Tools" on page 177
- "Logging" on page 182
- "Communication Problems Between Management Server and HTTPS Agents" on page 182
- "Certificate Deployment Problems" on page 196
- "Certificate Backup and Recovery in HPOM" on page 197

For more information on troubleshooting HTTPS agents, see the HP Operations Agent documentation.

Troubleshooting HTTPS-based Communication

If communication between an HP Operations management server and an HTTPS agent appears to be interrupted, for example, messages do not arrive at the Message Browser, or software or instrumentation is not distributed, execute the appropriate troubleshooting steps as described in the following sections.

Before you continue with the described actions, you should be familiar with the new HTTPS agent and the underlying communication concepts such as certificates.

This guideline describes possible actions to identify and solve HTTPS communication problems between HP Operations management servers, Certificate Authority Servers and managed nodes.

It is assumed, that the HPOM agent software is installed, but there is a problem in the communication between HP Operations managed nodes and HP Operations management servers in one or both directions.

In most installations, the HP Operations management server and Certificate Authority servers are installed on the same system.

Troubleshooting problems encountered with the communication between an HP Operations management server and an HTTPS agent is split into the following areas:

- "Troubleshooting Tools" on page 177
- "Logging" on page 182

Troubleshooting Tools

Ping an HTTPS-Based Application

HTTPS-based applications can be pinged to test if the application is active and responding. A ping may be executed against an application whether or not it has SSL enabled.

The `bbcutil` utility supports a `-ping` command line parameter that can be used to ping an HP Operations HTTPS-based application.

Use the following command on an HP Operations management server to ping a specified HTTPS-based application:

```
<OvInstallDir>/bin/bbcutil -ovrg server –ping [<hostname_or_ip_addr>] [count]
```

For example:

HTTP	`bbcutil -ovrg server -ping http://...`
HTTPS	`bbcutil -ovrg server -ping https://...`

Checks whether the communication service on the managed node specified by *<hostname_or_ip_addr>* is alive. If the hostname or IP address is omitted, localhost is assumed. An optional loop count can be specified after the hostname or IP address which causes the ping command to be repeated by the number of times specified.

For details about the command line parameters, see the *bbcutil* manual page.

In general, all `bbcutil` calls from an HP Operations management server to a managed node should include the `-ovrg server` parameter. For example:

```
bbcutil -ovrg server -ping https://...
```

If the HP Operations management server is a stand-alone system, the `-ovrg server` parameter maybe omitted. However, if the HP Operations management server is installed on an HA cluster, the `-ovrg server` parameter is required because a managed node certificate and a server certificate including two `OvCoreIds` are installed on each HP Operations management server. While on stand-alone systems, the managed node certificate and server certificate, including the `OvCoreIds`, are identical, they differ on cluster installations. The agent is only aware of the management server `OvCoreId`. It is not aware of the `OvCoreId` value of the management server.

Display the Current Status of an HTTPS-Based Application

An HTTPS-based application at a specified location can be requested to display its current status.

Use the following command to query a specified application:

`bbcutil -status` *<hostname_or_ip_addr:port>*

Queries the communication server located at the hostname and port specified by *<hostname_or_ip_addr:port>* for details about the current state of the server.

For details about the command line parameters, see the *bbcutil* manual page. If a port is not specified, the port number of the Communication Broker is used.

Display All Applications Registered to a Communication Broker

The Communication Broker at a specified location can be requested to display all applications that are registered to it.

Use the following command to list all applications that are registered to the specified Communication Broker:

`bbcutil -registrations|-reg` *<hostname_or_ip_addr>*

Queries a Communication Broker on the managed node specified by *<hostname_or_ip_addr>* and displays a list of all registered applications. If the hostname or IP is omitted, localhost is assumed.

For details about the Communication Broker command line parameters, see the *bbcutil* manual page.

Use What String

All executables contain a detailed UNIX-style `what` string that can be used to determine the precise version of the HTTPS-based communication software installed. Microsoft Windows executables also contain standard property strings.

List All Installed HP BTO Software Filesets on an HTTPS Managed Node

The `ovdeploy` tool can be used to list the installed HP BTO Software products and components. The following three levels of information can be displayed:

- Basic inventory
- Detailed inventory
- Native inventory

The following sections illustrate how to list the inventory and show examples of the output.

Basic Inventory

To display basic inventory information, enter the following command on an HP Operations management server:

```
ovdeploy -ovrg server -inv -host <hostname>
```

For example:

```
ovdeploy -ovrg server -inv -host hp_System_002
```

```
NAME                                        VERSION    TYPE
ARCHITECTURE
HP OpenView HTTP Communication              05.00.070  package
Windows 4.0 5.0 5.1 5.2
HP OpenView Deployment                      02.00.070  package
Windows 4.0 5.0 5.1 5.2
HP OpenView Security Certificate Management  01.00.070  package
Windows 4.0 5.0 5.1 5.2
HP OpenView Security Core                   02.00.070  package
Windows 4.0 5.0 5.1 5.2
...
```

Detailed Inventory

To display detailed inventory information, enter the following command on an HP Operations management server:

```
ovdeploy -ovrg server -inv -all -host <hostname>
```

For example:

```
ovdeploy -ovrg server -inv -all -host hp_System_002
```

```
<?xml version='1.0' encoding='UTF-8' standalone='yes'?> <inventory
    xmlns=">http://openview.hp.com/xmlns/depl/2003/inventory">
  <host>hpspi002.bbn.hp.com</host>
  <date>Thursday, October 30, 2003 12:24:48 PM</date>
  <package>
    <name>HP OpenView HTTP Communication</name>
    <version>05.00.070</version>
    <systemtype>IA32</systemtype>
    <ostype>Windows</ostype>
    <osvendor>MS</osvendor>
    <osversion>4.0 5.0 5.1 5.2</osversion>
    <osbits>32</osbits>
    <nativeinstallertype>msi</nativeinstallertype>
  </package>
  <package>
    <name>HP OpenView Deployment</name>
    <version>02.00.070</version>
    <systemtype>IA32</systemtype>
...
```

Native Inventory

To display native inventory information, enter the following command an HP Operations management server:

```
ovdeploy -ovrg server -inv -it native -host <hostname>
```

For example:

```
ovdeploy -ovrg server -inv -it native -host hp_System_002
```

```
NAME                                    VERSION
WebFldrs XP                             9.50.5318
HP OpenView Core Library                2.50.70
HP OpenView Certificate Management Client  1.0.70
HP OpenView HTTP Communication          5.0.70
ActivePerl 5.6.1 Build 633              5.6.633
HP OpenView Deployment                  2.0.70
Microsoft FrontPage Client - English    7.00.9209
```

Standard TCP/IP Tools

If SSL is not enabled, standard TCP/IP tools such as telnet can be used to contact HTTPS-based application. To use telnet to ping an HTTPS-based application execute the following commands:

Two carriage returns are required after the `PING` input line to telnet.

To end the telnet session, enter **`control-D`** and Return:

```
telnet <host> <port>
PING /Hewlett-Packard/OpenView/BBC/ping HTTP/1.1
```

The output takes the following form:

```
HTTP/1.1 200 OK
content-length: 0
content-type: text/html
date: Thu, 08 Aug 2008 08:20:24 GMT
senderid: fd7dc9c4-4626-74ff-9e5a09bffbae
server: BBC X.05.00.01.00; ovbbccb 05.00.100
```

`HTTP status 200 OK` indicates the HTTPS-based application has recognized the request and successfully responded. Other status may indicate a failure in the request or other error.

For a list of error codes, refer to:

`http://www.w3.org/Protocols/rfc2616/rfc2616.html`

RPC Calls Take Too Long

If an RPC call takes longer than the default timeout of 5 minutes, the following error messages may be displayed, for example, for a policy installation:

```
ERROR:   General I/O exception while connecting to host '<hostname>'.
          (xpl-117) Timeout occurred while waiting for data.
```

or

```
ERROR:  The Configuration server is not running on host '<hostname>'.
         Check if the Configuration server is in state running.
         (bbc-71) There is no server process active for address:
         https://<hostname>/com.hp.ov.conf.core/bbcrpcserver
```

This may happen if 1000 policies are installed using the `PolicyPackage` interface from `OvConf` or if the connection or target-machine is slow.

To prevent this the communication timeout (response timeout) can be changed using the following commands with the required time out value:

On the target system:

```
ovconfchg -ns bbc.cb -set RESPONSE_TIMEOUT <seconds>
```

On the HP Operations management server:

```
ovconfchg -ovrg server -ns bbc.http.ext.conf -set RESPONSE_TIMEOUT <seconds>
```

Note: The `RESPONSE_TIMEOUT` parameter must be set on both managed nodes.

A similar situation can arise when running any command that takes over 5 minutes to complete. The timeouts should be extended as follows.

On the managed node enter the commands:

Note: The unit is milliseconds in the second case.

```
ovconfchg -ns bbc.cb -set RESPONSE_TIMEOUT <seconds>
ovconfchg -ns depl -set CMD_TIMEOUT <milliseconds>
```

On the HP Operations management server, enter the command:

```
ovconfchg -ovrg server -ns bbc.http.ext.depl -set RESPONSE_TIMEOUT <seconds>
```

Logging

Errors in violation of security rules are recorded in a log file. For HTTPS-based servers, all client access can be additionally logged, if enabled.

To enable logging of all client access, set the following parameter value using the command:

```
ovconfchg -ns bbc.cb -set LOG_SERVER_ACCESS true
```

This will log all access to the Communication Broker. To view the logs, open one of the following files:

- *<OvDataDir>*`/log/System.txt` (ASCII)
- *<OvDataDir>*`/log/System.bin` (Binary)

You can additionally log access to all HP Communication Broker servers using the command:

```
ovconfchg -ns bbc.http -set LOG_SERVER_ACCESS true
```

You can additionally log all client access to the configuration and deployment application using the command:

```
ovconfchg -ns bbc.http.ext.conf -set LOG_SERVER_ACCESS true
```

Communication Problems Between Management Server and HTTPS Agents

The most likely areas where communication problems may be experienced are divided into the following sections:

- "Network Troubleshooting Basics" on page 182
- "HTTP Communication Troubleshooting Basics" on page 184
- "Authentication and Certificates Troubleshooting for HTTP Communication" on page 188
- "HPOM Communication Troubleshooting" on page 192

Network Troubleshooting Basics

Basic network troubleshooting uses the following commands:

ping	*<SYSTEMPATH>*/ping
nslookup	*<SYSTEMPATH>*/nslookup
telnet	*<SYSTEMPATH>*/telnet
ovgethostbyname	*<INSTALLDIR>*/bin/ovgethostbyname (for use on Solaris systems only in place of nslookup)

Note: The actions described below may not work if communication between an HP Operations management server or Certificate Authority server and HP Operations managed node has to pass:

- Firewalls
- NATs
- HTTP Proxies

Contact your Network Administrator for more information.

To check for basic network problems, complete the following steps:

1. Check if the name resolution for the HP Operations management server, Certificate Authority server and HP Operations managed node is consistent on all affected systems.

   Use ping, and nslookup (on Solaris: ovgethostbyname) with the Fully Qualified Domain Name (FQDN) on all systems with all systems as targets.

   ```
   bbcutil -gettarget <nodename>
   ```

2. Check if all systems (HP Operations management server, Certificate Authority server and managed node) are accessible.

   Use one of the following commands:

   - *<OvInstallDir>*/bin/bbcutil -ping *<FQDN>*
   - telnet *<FQDN>*

3. Check if HTTP communication is working by using a Web browser to connect to the Communication Broker. The Communication Broker, ovbbccb, must be running for this check.

   To retrieve the assigned *<AGENT-BBC-PORT>* value, enter the command:

   ```
   bbcutil -getcbport <agenthostname>
   ```

   For example, if you enter the command:

   ```
   bbcutil -getcbport mysystem.mycom.com
   ```

   the following output is displayed:

   ```
   mysystem.mycom.com:8008
   ```

   On the HP Operations management server, open a Web browser and type the following URL:

`http://<HPOM managed node>:<AGENT-BBC-PORT>/Hewlett-Packard/OpenView/BBC/`

The default port number for *<AGENT-BBC-PORT>* is 383.

Repeat this step from the managed node to the HP Operations management server:

`http://<HPOM management server>:<AGENT-BBC-PORT>/Hewlett-Packard/OpenView/BBC/`

The `HP OpenView BBC Information Modules` page should appear and allow you to check `ping` and `status` or list registered services and HPOM resource groups (ovrg).

HTTP Communication Troubleshooting Basics

Basic HTTP communication troubleshooting uses the following commands:

ovc	*<INSTALLDIR>*/bin/ovc
ovconfget	*<INSTALLDIR>*/bin/ovconfget
ovbbccb	*<INSTALLDIR>*/bin/ovbbcutil
ps	*<SYSTEMPATH>*/ps

Note: Even if the communication between HP Operations management server or Certificate Authority server and managed node has to pass:

- Firewalls
- NATs
- HTTP Proxies

the following actions must work. If they do not, contact your Network Administrator for more information.

Note: If the communication between HP Operations management server or Certificate Authority server and managed node is not allowed to pass through the firewalls, one or more HTTP Proxies must be used (see the corresponding sections).

To check for HTTP communication problems, complete the following steps:

1. On all systems, the HP Operations management server, Certificate Authority server and managed node, check if:

   The HP Communication Broker `ovbbccb` is running with the following commands:

   ```
   ovc -status
   ```

   The `ovbbccb` process must be listed as running. The output is as follows:

   ```
   ovcd      OV    Control               CORE      (2785)  Running
   ovbbccb   OV    Communication Broker   CORE      (2786)  Running
   ```

```
ovconfd   OV    Config and Deploy      CORE      (2787)  Running
ovcs      OV    Certificate Server     SERVER    (3024)  Running
coda      OV    Performance Core       AGENT     (2798)  Running
opcmsga   OMU   Message Agent          AGENT,EA  (2799)  Running
opcacta   OMU   Action Agent           AGENT,EA  (2800)  Running
opcmsgi   OMU   Message Interceptor    AGENT,EA  (2801)  Running
opcle     OMU   Logfile Encapsulator   AGENT,EA  (2805)  Running
opcmona   OMU   Monitor Agent          AGENT,EA  (2806)  Running
opctrapi  OMU   SNMP Trap Interceptor  AGENT,EA  (2810)  Running
```

`ps <OPT> | grep ovbbccb`

`ovbbccb` must be listed.

`<OvInstallDir>/bin/bbcutil -status`

The status of `ovbbccb` must be `ok`.

Note: Make a note of the ports listed using the command:

`bbcutil -getcbport <hostname>`

- on managed node as `<AGENT-PORT>`
- on management server as `<MGMT-SRV-PORT>`
- on Certificate Authority server as `<CA-SRV-PORT>`

Alternatively, you can use the command:

`ovconfget bbc.cb.ports PORTS`

You can start the Communication Broker with the command:

`ovc -start`

No error messages should be displayed.

If the `ovbbccb` process is not running:

a. Check the logfile for error messages in the appropriate file:
   - `<OvDataDir>/log/System.txt` (ASCII)
   - `<OvDataDir>/log/System.bin` (Binary)

b. Start the Communication Broker with the command:

   `<OvInstallDir>/bin/bbcutil -nodaemon -verbose`

   If there is any problem, errors are displayed in detail at startup. The port number it uses is also displayed on startup.

c. For more detailed output use the command:

   `OVBBC_TRACE=true <OvInstallDir>/bin/bbcutil -nodaemon -verbose`

This displays a very significant amount of detailed information. This detail can also be obtained using HPOM tracing.

2. Check the configuration of the Communication Broker port settings with the following commands:
   a. Lists all Communication Broker ports:

      ```
      bbcutil -getcbport <hostname>
      ```

   b. Check if the default `DOMAIN` parameter is correctly set for the managed nodes using the command:

      ```
      ovconfget bbc.http DOMAIN
      ```

      This should be set to the default domain, for example, `myco.com`. This parameter may be used to find a match for the parameters configured in step 2.a above.

   c. Check if a process has the Communication Broker port open and is listening for connections using the command:

      ```
      netstat -an | grep \.383
      ```

      You should see something similar to (varies on each platform):

      ```
      tcp        0        0  *.383        *.*        LISTEN
      ```

      `LISTEN` verifies that a process is listening on the specified port. If this is displayed and the Communication Broker is not running, another process is using the port and the Communication Broker will not startup. This can be verified with steps 1.a and 1.b.

3. Check the HTTP Communication capabilities by entering the following commands:
   - On the HP Operations management server and the Certificate Authority server:

     ```
     <OvInstallDir>/bin/bbcutil -ovrg server -ping http://<HPOM managed node>
     [:<AGENT-PORT>]/
     ```

   - On the managed node:

     ```
     <OvInstallDir>/bin/bbcutil -ping http://HPOM management server[:<MGMT-SRV-
     PORT>]/
     ```

     ```
     <OvInstallDir>/bin/bbcutil -ping http://Certificate Authority server[:<CA-
     SRV-PORT>]/
     ```

   Note: If no port is specified in these command, the default port 383 is used.

   Each call should report:

   ```
   status=eServiceOK
   ```

4. Check if the managed nodes have the correct Communication Broker port configuration. Do not specify a port number in the URI. OV communication must be able to resolve the Communication Broker port number on its own. If the ping works with the port number, but does not work without the port number, the local managed node is not correctly configured. Go back to step 2.

5. Check if the HTTP Proxy is correctly configured using the command:

   ```
   bbcutil -gettarget <nodename>
   ```

   For example, if you enter the command:

   ```
   bbcutil -gettarget mysystem.mycom.com
   ```

   Output of the following form is displayed:

   ```
   Node: mysystem.mycom.com:8008 (14.133.123.10)
   ```

   If a proxy is configured, it will be displayed.

   For example, if you enter the command:

   ```
   bbcutil -gettarget www.mycom.com
   ```

   Output of the following form is displayed:

   ```
   HTTP Proxy: web-proxy:8008 (14.193.1.10)
   ```

   ```
   ovconfget bbc.http PROXY
   ```

   Although not recommended, applications may set their own private PROXY setting. The above setting is valid for the whole managed node. An individual application may override this value in its own private namespace:

   If the *<comp id>* or *<appname>* is not known, check using `ovconfget` the entire configuration for all proxy settings in the namespaces starting with:

   ```
   ovconfget bbc.http.ext.<comp id>.<appname>
   ```

   ```
   bbc.http.ext
   ```

6. Check on the HP Operations management server and the Certificate Authority server systems that the proxy is working and supports the `CONNECT` command.

   Note: The blank lines are important.

   On some platforms, it may not be possible to echo commands typed into telnet.

   Enter the command:

   ```
   telnet <proxy> <proxy port>
   CONNECT <AGENT>:<AGENT PORT> HTTP/1.0

   PING /Hewlett-Packard/OpenView/BBC/ HTTP/1.0
   ```

   To exit telnet, enter **Control-D**.

   The output should be similar to the following. If the Communication Broker is up and running on the target managed node, the HTTP status should be `200 OK` .

   ```
   HTTP/1.1 200 OK
   cache-control: no-cache
   content-type: text/html
   date: Fri, 06 Feb 2004 15:15:02 GMT
   senderid: fd7dc9e4-4626-74ff-084a-9e5a09bffbae
   server: BBC 05.00.101; ovbbccb 05.00.101HP OpenView BBC Information Modules:
   ```

```
Node:         ping.bbn.hp.com
Application:  ovbbccb
Version:      05.00.101
Modules:      ping
              status
              services
              ovrg

Connection closed by foreign host.
```

7. Check on the HP Operations managed node that the proxy is working and supports the `CONNECT` command.

   Note: The blank lines are required. On some platforms, it may not be possible to echo commands typed into telnet.

   Enter the command:

   ```
   telnet <proxy> <proxy port>
   CONNECT <MGMT-SRV>:<MGMT-SRV PORT> HTTP/1.0

   PING /Hewlett-Packard/OpenView/BBC/ HTTP/1.0
   ```

   or

   ```
   telnet <proxy> <proxy port>
   CONNECT <CA-SRV>:<CA-SRV PORT> HTTP/1.0

   PING /Hewlett-Packard/OpenView/BBC/ HTTP/1.0
   ```

   To exit telnet, enter **Control-D**.

   See the previous point for a sample output.

8. Enable logging for HTTP access to the Communication Broker.

   ```
   ovconfchg -ns bbc.cb -set LOG_SERVER_ACCESS true
   ```

   This will log all access to the Communication Broker. To see the logs, use:

   ```
   ovlogdump <OvDataDir>/log/System.txt
   ```

   You can additionally log access to all HP Operations servers using:

   ```
   ovconfchg -ns bbc.http -set LOG_SERVER_ACCESS true
   ```

Authentication and Certificates Troubleshooting for HTTP Communication

Troubleshooting Basic HTTP communication uses the following commands:

ovc	*<INSTALLDIR>*/bin/ovc
ovconfget	*<INSTALLDIR>*/bin/ovconfget

ovconfchg	*<INSTALLDIR>*/bin/ovconfchg
ovcoreid	*<INSTALLDIR>*/bin/ovcoreid
ovcert	*<INSTALLDIR>*/bin/ovcert
bbcutil	*<INSTALLDIR>*/bin/bbcutil

To check for authorization and certificate related HTTP communication problems, complete the following steps:

1. Check the OvCoreID of each system.

   On the HP Operations management server or the Certificate Authority server, enter the command:

   ```
   ovcoreid -ovreg server
   ```

   On the managed node, enter the command

   ```
   ovcoreid
   ```

   Make a note of each of the displayed OvCoreID values:

   - *<MGMT-SRV-COREID>*
   - *<CA-SRV-COREID>*
   - *<AGENT-COREID>*

2. Check the certificates on the HP Operations management server or Certificate Authority server and on managed node using the following command:

   ```
   ovcert -list
   ```

   Note: There are 3 certificates on the HP Operations management server system or Certificate Authority system:

   - HP Operations management server certificate
   - Certificate authority certificate
   - Managed node certificate

   When an HP Operations management server is installed on a cluster (high availability environment), the certificates of the HP Operations management server and the agent on the management server are not the same. On non-cluster installations, the certificates must be identical.

   On each system there must be at least following Certificates.

   On the managed node:

   ```
   | Certificates:                          |
   ```

```
|     <AGENT-COREID>                  (*)      |
```

On the management server or the Certificate Authority server:

```
| Certificates:                                |
|     <MGMT-SRV-COREID>|<CA-SRV-COREID>   (*)      |
```

On all systems:

```
| Trusted Certificates:                        |
|     <CA-SRV-COREID>                          |
```

Note: The `(*)` signifies that the private key for the certificate is available.

If one of the certificates is missing, see "Working with Certificates" on page 134 and generate the required certificates.

To get more detailed info about the installed certificates, use the following commands:

On the managed node:

```
ovcert -check
```

On the management server:

```
ovcert -check -ovrg server
```

An example of the output is shown below:

```
OvCoreId set                      : OK
Private key installed              : OK
Certificate installed              : OK
Certificate valid                  : OK
Trusted certificates installed     : OK

Check succeeded.
```

To check that the installed certificates are valid, use the following command and make sure that the current date is between the `valid from` and `valid to` dates of the installed certificates:

```
ovcert -certinfo <CertificateID>
```

Note: The CertificateID of a trusted certificates is the OvCoreID of the certificate server prefixed with a `CA_`.

An example of the output is shown below:

```
# ovcert -certinfo 071ba862-3e0d-74ff-0be4-b6e57d0058f2

Type       :  X509Certificate
Subject CN :  071ba862-3e0d-74ff-0be4-b6e57d0058f2
Subject DN :  L:  alien2.ext.bbn.com
              O:  Hewlett-Packard
```

```
              OU: OpenView
              CN: 071ba862-3e0d-74ff-0be4-b6e57d0058f2
Issuer CN  :  CA_99300c4e-f399-74fd-0b3d-8938de9900e4
Issuer DN  :  L:  tcbbn054.bbn.hp.com
              O:  Hewlett-Packard
              OU: OpenView
              CN: CA_99300c4e-f399-74fd-0b3d-8938de9900e4
Serial no. :  04
Valid from :  01/27/04 12:32:48 GMT
Valid to   :  01/22/24 14:32:48 GMT
Hash (SHA1):  60:72:29:E6:B8:11:7B:6B:9C:82:20:5E:AF:DB:D0: ...
```

Note: An HTTPS agent is also installed on an HP Operations management server system.

If calling `ovcert -list` on a management server system, you are given the certificate details of the agent on the management server system as well as the details of the certificate for the management server and the CA.

3. Check the HTTPS communication capabilities using the following commands.

Note: The following actions must work even if communication between an HP Operations management server or a Certificate Authority server and an managed node has to pass:

- Firewalls
- NATs
- HTTP Proxies

If they do not, contact your Network Administrator for more information.

Note: If the communication between HP Operations management server or Certificate Authority server and HP Operations managed node is not allowed to pass through the firewalls, one or more HTTP Proxies must be used (see the corresponding sections).

On an HP Operations management server or Certificate Authority server:

`<OvInstallDir>/bin/bbcutil -ovrg server -ping https://<HPOM managed node name>[:<AGENT-PORT>]/`

On a managed node:

`<OvInstallDir>/bin/bbcutil -ping https://<HPOM management server name>[:<MGMT-SRV-PORT>]/`

`<OvInstallDir>/bin/bbcutil -ping https://Certificate Authority server[:<CA-SRV-PORT>]/`

Each call should report:

`status=eServiceOK`

The reported OvCoreID must match with the OvCoreIDs that you noted in the first step:

`coreID=<COREID>`

HPOM Communication Troubleshooting

Troubleshooting HPOM communication uses the following commands:

ovc	*<INSTALLDIR>*/bin/ovc
ovconfget	*<INSTALLDIR>*/bin/ovconfget
ovconfchg	*<INSTALLDIR>*/bin/ovconfchg
ovcoreid	*<INSTALLDIR>*/bin/ovcoreid
ovpolicy	*<INSTALLDIR>*/bin/ovpolicy
ovcs	*<INSTALLDIR>*/bin/ovcs
opcagt	*<INSTALLDIR>*/bin/OpC/opcagt
opcragt	*<INSTALLDIR>*/bin/OpC/opcragt
opccsa	*<INSTALLDIR>*/bin/OpC/opccsa
opcssam	*<INSTALLDIR>*/bin/OpC/opccsam
opcsv	*<INSTALLDIR>*/bin/OpC/opcsv
opcnode	*<INSTALLDIR>*/bin/OpC/opcnode
opc	/usr/bin/OpC/opc

To check for HPOM communication problems, complete the following steps:

1. HP Operations managed nodes must be in the node bank.
2. The Fully Qualified Domain Name (FQDN) of the HP Operations managed node must match.
3. The communication type of the managed node must be `HTTPS`.
4. The `OvCoreID` of the managed node must match.

   Check the value of the managed node `OvCoreID` stored in the HPOM database using the command:

   `opcnode -list_id node_list=<HPOM managed node>`

   It must match the *<AGENT-COREID>*.

   To check, on the managed node call the command:

   <OvInstallDir>/bin/ovcoreid

You can change the managed node OvCoreID from the HP Operations management server using the command:

```
opcnode -chg_id node_name=<HPOM managed node> id=<AGENT-COREID>
```

You can change the OvCoreID on the managed node using the command:

```
ovcoreid -set <NEW-AGENT-COREID>
```

Note: Changing the OvCoreId of a system is an operation that must be done with great care because it changes the identity of a managed node. All managed node-related data, such as messages, are linked by the OvCoreId of a managed node. Changing the value of the OvCoreID should only be executed by experienced users who know exactly what they want to do and what is being affected by attempting this change, especially on the HP Operations management server.

Sometimes multiple systems use the same OvCoreId. To solve this problem, see the HP Operations agent documentation.

5. Check that all HP Operations management server processes are running using the commands:

   ```
   opcsv -status
   ```

   All registered processes must be in the state running.

   ```
   ovc -status
   ```

   All registered core processes must be in state running.

6. Make sure that the operator is responsible for the:

   - HP Operations managed node and its node group
   - Message group

   Reload the Message Browser.

7. Check for pending certificate requests.

   On the Certificate Authority server enter the command:

   ```
   opccsa -list_pending_cr
   ```

   Check if the managed node is listed by nodename, IP address or OvCoreID and whether all parameters are consistent.

   Manually grant pending certificate requests with the command:

   ```
   opccsa -grant <NODE>|<Certificate_Request_ID>
   ```

   If the parameter are not consistent, change the values on the HP Operations management server and managed node, as required.

   On the HP Operations managed node, stop and restart all processes with the commands:

   ```
   ovc -kill
   ```

Verify that all processes are stopped with the command:

```
ps <OPT> | grep /opt/OV
```

```
ovc -start
```

Note: To manually trigger a Certificate Request, first check that there is no certificate already installed with the command:

```
ovcert -status
```

If no certificate is installed, enter the command:

```
ovcert -certreq
```

The `ovcd` process of the HTTPS agent must be running for the `ovcert -certreq` call to work. Certificate requests are automatically sent during agent startup, so just the agent startup is sufficient, unless the `CERTIFICATE_DEPLOYMENT_TYPE` is set to `Manual`. This is done with the command:

```
ovconfchg -ns sec.cm.client -set CERTIFICATE_DEPLOYMENT_TYPE Manual
```

Therefore, the `ovcert -certreq` command is only of interest if `Manual` certificate deployment type is chosen, or if the certificate was removed while the agent was running. For example, no `ovc -kill` command run before removing the certificate.

If a certificate is already installed, the following error message is displayed:

```
ERROR: (sec.cm.client-125) There is already a valid certificate for this
node installed.
```

8. If there are no managed node messages in the Message Browser on a managed node, execute the following checks:
   - Check if all processes are running:

     ```
     ovc -status
     ```

     All registered processes must be running and no process should run twice.
   - Check if the expected policies are deployed:

     ```
     ovpolicy -list
     ```
   - Check the `MANAGER`, `MANAGER_ID`, and `CERTIFICATE_SERVER` settings:

     ```
     ovconfget sec.cm.client CERTIFICATE_SERVER
     ```

     This must match the Certificate Authority server.

     ```
     ovconfget sec.core.auth MANAGER
     ```

     This must match the HP Operations management server.

     ```
     ovconfget sec.core.auth MANAGER_ID
     ```

     This must match the OvCoreID of the HP Operations management server.

To check the OvCoreId of the management server, on the management server enter the command:

```
ovcoreid -ovrg server
ovconfget eaagt OPC_PRIMARY_MGR
```

This setting is optional, but when set, it must match the HP Operations management server.

Note: If the HP Operations management server is not the primary manager, additional checks have to be performed.

The HP Operations management server must appear with consistent values in the file:

<OvDataDir>/datafiles/policies/mgrconf/*<ID>*_data

- Check the settings of message suppression.
- Check the settings of message buffering.
- Check if the message buffer file is growing:

  ls -l *<OvDataDir>*/tmp/OpC/msgagtdf

  or on HP Operations management server:

  opcragt -status *<nodename>*
- Send a message to be forwarded to the server:

  opcmsg a=appl o=object msg_t=*<my_text>*
- Check if messages appear in the message manager queue file:

  strings /var/opt/OV/share/tmp/OpC/mgmt_sv/msgmgrq | grep *<my_text>*

9. If DEPLOYMENT, ACTIONS or HBP to a managed node fails, on the managed node, check the status of the agent with the command:

   opcragt -status

   If this reports no problems, the problem is not HTTPS communication dependent.

HTTPS Communication and Time Zones

ovbbccb provides increased security on UNIX operating systems by using a feature known as chroot (). A chroot on UNIX operating systems is an operation which changes the root directory. Whenever the ovbbccb process starts up, it is rooted to *<OvDataDir>* in UNIX. This ensures that it can access files only under *<OvDataDir>*. It cannot access any other files.

For time zone conversions, the system files for time zone are needed, which are located in a now inaccessible directory. ovbbccb cannot access the time zone file and writes the date information in UTC(GMT) format rather than actual time zone set for the system.

To establish the correct time zone, create a similar directory structure under *<OvDataDir>* as is available under .../zoneinfo/*<TZ>* and copy the actual time zone file:

1. Stop all the HPOM processes:

   /opt/OV/bin/ovc -kill

2. Check the /etc/time zone file for the current time zone (TZ value), for example:

   TZ=US/Eastern

3. Create the following directory based on the TZ value:

   mkdir -p *<OvDataDir>*/usr/share/lib/zoneinfo/*<TZ>*

   If the TZ value contains entries separated by a /, as in our example with TZ=US/Eastern, create the directory structure up to the last slash:

   mkdir -p *<OvDataDir>*/usr/share/lib/zoneinfo/US

   Note: Substitute *<OvDataDir>* with path used by the managed node platform. For details see the HP Operations agent documentation.

   For example: *<OvDataDir>* on Solaris is: /var/opt/OV

   Make sure that the directory structure under *<OvDataDir>* is exactly same as that of /usr/share/lib/zoneinfo/<TZ>.

4. Copy the time zone resource file to the newly created directory:

   cp /usr/share/lib/zoneinfo/*<TZ>*<*OvDataDir*>/usr/share/lib/zoneinfo/*<TZ>*

   On HP-UX systems, also copy the following file:

   usr/lib/tztab

5. Start all the HPOM processes:

   /opt/OV/bin/ovc -start

All the messages subsequently logged by ovbbccb should have the correct timestamp.

Certificate Deployment Problems

During certificate deployment, the situation may arise that there are two pending certificate requests for the same managed node in the Certificate Server Adapter's list of pending certificate requests.

For example, this can occur if the certificate request is triggered from the managed node. This certificate request is not granted and remains pending in the Certificate Server Adapter's internal list. If you now deinstall the agent software and re-install it, another certificate request is triggered. The new request also contains a new OvCoreID, because re-installing the managed node generates a new OvCOreID. This certificate also remains in the list of pending certificate requests.

The listing of the pending certificate requests also contain a time stamp of when the certificate request was received by the HP Operations management server. It is clear which certificate request is newer and valid. Grant the newest one and remove any older requests.

Alternatively, there are two further ways of removing unwanted certificate requests:

- Log in as an HPOM administrator and remove all certificate requests for a “problematic” managed node and then issue a new certificate request from the managed node with the command:

  `ovcert -certreq`

  > **Note:** The `ovcd` process of the HTTPS agent must be running for the `ovcert -certreq` call to work. Certificate requests are automatically sent during agent startup, so just the agent startup is sufficient, unless the `CERTIFICATE_DEPLOYMENT_TYPE` is set to `Manual`.
  >
  > Therefore, the `ovcert -certreq` command is only of interest if `Manual` certificate deployment type is chosen, or if the certificate was removed while the agent was running. For example, no `ovc -kill` command run before removing the certificate.

  This results in a single certificate request for the managed node which can then be mapped and granted in the usual way. See "Working with HTTPS Managed Nodes" on page 157.

- If as administrator, you cannot execute the `ovcert -certreq` command on the managed node and so cannot issue a new certificate request, then retrieve the valid `OvCoreID` from the managed node by executing the command:

  `<OvInstallDir>/bin/bbcutil -ovrg server -ping <nodename>`

  List all certificate requests and grant the certificate request that contains valid `OvCoreID` and remove any others.

Certificate Backup and Recovery in HPOM

It is extremely important to be aware of the impacts of losing a private key or when keys and certificate errors arise. The normal configuration upload and download does not include certificate and key data.

There is a utility on the HP Operations management server to back up and recover certificates plus the associated private keys and OvCoreIds:

`/opt/OV/bin/OpC/opcsvcertbackup/`

This utility has the following options:

- **`-remove`**

  Removes all certificates from an HP Operations management server, including:

  - Certificate Authority root certificate and its private key.
  - Server certificate and its private key.

- Managed node certificate on the HP Operations management server.

However, a backup is also created automatically before the removal takes place.

- **-backup**

  A tar archive is created at the following default address:

  `/tmp/opcsvcertbackup.<date_time>.tar`

  The *<date_time>* format is `YYMMDD_hhmmss`.

  The default storage location can be changed by using the **-file** option.

  The information recorded includes:

  - Certificate Authority root certificate, private key and ID
  - HP Operations management server certificate with key and OvCoreId
  - Managed node certificate with key and `OvCoreId`

  You must secure the data by using the **-pass** option with a password.

  The tar archive contains a text file named:

  `opcsvcertbackup.<date_time>.txt`

  This information can be useful for archiving and includes OvCoreIds of the backed up certificates, hostname, and time stamp of the backup. This information is not used during a restore.

- **-restore**

  A tar archive as created using the `-backup` option can be restored using this command.

  The filename must be provided with the `-file` option. The password used at backup time must be entered with the `-pass` option.

  The restore cannot work, if any of the certificates or private keys for the Certificate Authority, HP Operations management server, or managed node already exists on the management server system but are not the same as the corresponding values stored in the backup archive.

  To avoid this, enforce the restore by using the **-force** option. `opcsvcertbackup` also returns with an error when the OvCoreIds of the certificates to be restored do not fit with those stored in the HPOM database. When the `-force` option is used, the OvCoreIds are replaced and confirmation is displayed.

When to Back Up Certificates

The following are the times when a backup using `opcsvcertbackup` is recommended:

- **Initial HPOM Installation**

  After a successful HP Operations management server installation, it is highly recommended to make a backup of the certificate data with the command:

```
opcsvcertbackup -backup
```

The resulting tar archive should be stored in a secure place.

- **HP Operations Management Server Re-installation on Alternative System**

  Perform a standard HP Operations management server installation on the alternative system. Install the backup from the original management server installation onto the newly installed system with the command:

  ```
  opcsvcertbackup -restore –file <filename> –pass <password> –force
  ```

  Note: The `-force` option must be used because the server installation has automatically created a Certificate Authority, HP Operations management server, and managed node certificates. These certificates are unsuitable because the managed nodes are configured to use the existing ones from the first installation.

- **Recovery**

  If something is deleted accidentally, use the command:

  ```
  opcsvcertbackup -restore –file <filename> –pass <password>
  ```

  Carefully check any error output.

- **Recovery from Configuration Errors**

  If a normal recovery without force option id not successful, check the error messages from the `opcsvcertbackup` call. If this does not help, clean the certificate information stuff with the command:

  ```
  opcsvcertbackup –remove
  ```

  or directly overwrite the existing certificate configuration with the command:

  ```
  opcsvcertbackup -restore –file <filename> –pass <password> –force
  ```

- **Configuring a Certificate Trust for Flexible Management Environments**

  After creating a certificate trust it is recommended that you make a new backup. This ensures that the additional root certificate(s) can be restored in case a recovery is needed.

- **Configuring a Shared Certificate Authority**

  When configuring a shared Certificate Authority, the following command can be useful for removing the unwanted certificates from a second HP Operations management server installation.

  ```
  opcsvcertbackup –remove
  ```

Tracing HPOM

This section describes how to manage tracing HPOM in the following sections:

- "Tracing Overview" on page 200
- "Using HP-Style Tracing for HPOM" on page 200

- "HPOM-Style Tracing" on page 203
- "Example of Tracing HPOM Processes" on page 206

Tracing Overview

There are two styles of tracing that can be used for HPOM:

- HP-style tracing

  HP tracing can be used to help solve problems with HTTPS agents and the HP Operations management server. For more information, see "Using HP-Style Tracing for HPOM" on page 200.
- HPOM-style tracing

  The configuration settings which specify the HPOM tracing are set with the `ovconfchg` command. For more information, see "HPOM-Style Tracing" on page 203.

Using HP-Style Tracing for HPOM

There are two ways to trace HPOM using HP tracing:

- Configure Remote Tracing Using the Windows Tracing GUI. For more information, see the HP Operations Agent documentation.
- Configure Manual HP Tracing Using Trace Configuration Files. For more information, see "Configuring Manual HP Tracing Using Trace Configuration Files" on page 200.

For the HP-style tracing overview, see the *HPOM Concepts Guide*.

Configuring Manual HP Tracing Using Trace Configuration Files

In many cases, in particular on UNIX systems, the simplest way is to manually create the trace configuration files specifying the components to be traced and log the trace output into a file. Three management server and three agent example trace configuration files are provided at the following location on the management server system:

`/opt/OV/contrib/OpC/TraceConfig`

You must copy the appropriate file to the managed node system, if you want to use it to trace an agent.

Note: You can also use the Tracing GUI to create a trace configuration file on a Windows system and then copy this to the system where you want to investigate a problem.

These files include trace configuration statements for all HPOM processes. Refer to the lines beginning with "`APP:`". If you want to trace specific processes, create a new trace configuration file and copy and paste the appropriate pieces from the example files and add the header line - the first line, beginning with `TCF`.

HP Tracing implements a hierarchy of elements starting with Applications, Components, Categories and Attributes. In HP Tracing terminology, the processes defined by OPC_TRC_PROCS and OPC_DBG_PROCS are referred to as Applications. The TRACE AREAS defined by the OPC_TRACE_AREA parameter are referred to as subcomponents.

Component = *<component name>*

Trace area = *<sub-component>*

Category = Trace

To configure the same type of trace configuration using HP Tracing, you create a Trace Configuration File, enable tracing using the ovtrccfg tool, and monitor the trace messages using the ovtrcmon tool.

Trace Configuration File

```
TCF Version 3.2
APP: "opcmsga"
SINK: Socket "prodnode" "node=10.1.221.22;"
TRACE: "eaagt.actn" "Trace" Info Warn Error Developer Verbose
TRACE: "eaagt.debug" "Trace" Info Warn Error Developer Verbose
TRACE: "eaagt.init" "Trace" Info Warn Error Developer Verbose
TRACE: "eaagt.msg" "Trace" Info Warn Error Developer Verbose
APP: "opcacta"
SINK: Socket "prodnode" "node=10.1.221.22;"
TRACE: "eaagt.actn" "Trace" Info Warn Error Developer Verbose
TRACE: "eaagt.init" "Trace" Info Warn Error Developer Verbose
TRACE: "eaagt.msg" "Trace" Info Warn Error Developer Verbose
```

Activating Tracing

To activate tracing into a local file, complete the following steps:

```
/opt/OV/support/ovtrcadm -a localhost
```

```
/opt/OV/support/ovtrccfg -server localhost <my_trace_config_file>
```

For example:

```
ovtrccfg -server localhost /opt/OV/contrib/OpC/TraceConfig/ServerAll.tcf
```

Viewing Trace Results

To view the trace output you need to use the formatting tool ovtrcmon:

```
/opt/OV/support/ovtrcmon -fromfile <binary_output> [ -tofile <ascii-output> ]
```

You can specify output formats. Details are available from the ovtrcmon usage text:

```
/opt/OV/support/ovtrcmon -help
```

An alternative way to capture trace output, assuming you want to use one of the pre-configured trace configuration files from the directory on the management server:

`/opt/OV/contrib/OpC/TraceConfig/*.tcf`:

is as follows:

1. In your trace configuration file (file extension `.tcf`), replace the lines beginning with `SINK: File` with the string:

   `SINK: Socket "localhost" "node=localhost;"`
2. Load the trace configuration file using the command:

   `/opt/OV/support/ovtrccfg <my_trace_config_file>`
3. Start `ovtrcmon` to dump the output into a file:

   `/opt/OV/ovtrcmon -server localhost > <my_ascii_trace_output_file>`

   For output formatting options, see the `ovtrcmon` usage message.

Disable Remote Tracing (No Ports Opened)

The `ovtrcd` process, by default, opens port 5053 for external access. You can switch off the opening of this externally visible port either on HPOM management server or on the managed node (see the HP Operations Agent documentation for the description of the method on managed nodes). To disable remote tracing on the HPOM management server, properly configure the `bbc_inst_defaults` file as follows.

The `bbc_inst_defaults` file on the management server contains the configuration setting:

`eaagt:DISABLE_REMOTE_TRACE_AT_INSTALL`

If this setting is set to `TRUE`, all appropriate newly installed agents automatically execute the following steps, as required by the above method, before they are started.

`ovtrcadm -disableremotetracing`

`/opt/OV/support/ovtrcadm -srvshutdown`

`/opt/OV/lbin/xpl/trc/ovtrcd`

For more information on how to configure the `bbc_inst_defaults` file, see the HP Operations agent documentation or the example file at the following location:

`/etc/opt/OV/share/conf/OpC/mgmt_sv/bbc_inst_defaults.sampl`

Switch Off Tracing

To switch off tracing, enter the command:

`/opt/OV/support/ovtrccfg off`

HPOM-Style Tracing

The configuration settings which specify the HPOM tracing are set with the `ovconfchg` command.

Activate HPOM-Style Tracing on the Management Server

You can activate the HPOM trace facility for the management server processes by entering the following command:

```
ovconfchg -ovrg server -ns opc -set OPC_TRACE TRUE
```

This entry is always required and enables tracing for the areas `MSG` and `ACTN`.

It is not necessary to restart any processes. Doing so may also remove the cause of the problem you are investigating.

Disabling HPOM-Style Tracing

To disable HPOM problem tracing on the HPOM management server, enter one of the following commands:

```
ovconfchg -ovrg server -ns opc -clear OPC_TRACE
```

or

```
ovconfchg -ovrg server -ns opc -set OPC_TRACE FALSE
```

To inform the processes about new configuration settings on the management server, enter the command:

```
/opt/OV/bin/OpC/opcsv -trace
```

Trace Output File Locations

Trace information is written to the `trace.bin` logfile on the HPOM management server:

<OvDataDir>`/share/tmp/OpC/mgmt_sv/trace.bin`

Default: `/var/opt/OV/share/tmp/OpC/mgmt_sv/trace.bin`

For more information about HPOM tracing on managed nodes, see HP Operations Agent documentation.

Configuring HPOM-Style Tracing of the Management Server

This reduces the amount of data that is entered into the trace output file and simplifies the interpretation of the trace logfile. You can activate tracing for specific functional areas by specifying one or more functional areas in the trace statement.

Functional Areas

You can select the most suitable functional areas from the following list to more precisely target the area of investigation. Functional areas are set using the `OPC_TRACE_AREA` statement.

Note: Not all functional areas are available for all processes.

`ACTN`	Actions.
`ALIVE`	Agent-alive check.
`ALL`	All tracing areas (except `DEBUG` and `PERF`).
`API`	Configuration API.
`AUDIG`	Auditing.
`DB`	Database.
`DEBUG`	Debugging information. Use this option carefully, as it provides extensive and detailed information, but the trace logfile will also be correspondingly large.
`DIST`	Distribution.
`INIT`	Initialization.
`INST`	Installation.
`INT`	Internal.
`LIC`	Licensing.
`MISC`	Miscellaneous.
`MSG`	Message flow.
`NAME`	Name resolution.
`NLS`	Native language support.
`NTPRF`	NTPerfMon.
`PERF`	Performance.
`SEC`	Security.
`SRVC`	Service.

Customize Tracing

To configure tracing:

1. Specify `OPC_TRACE TRUE`.

   This is always required and enables tracing for the areas `MSG` and `ACTN`.

2. To trace a specific functional areas, select the appropriate functional area or management server/agent process by entering statements of the following formats:

   ```
   OPC_TRACE_AREA <area>[,<area>]
   OPC_TRC_PROCS <process>[,<process>]
   OPC_DBG_PROCS <process>[,<process>]
   ```

<area>	HPOM area to be traced or debugged. By default, `MSG` and `ACTN` are enabled. For a list of all available areas, see the "Functional Areas" on page 204.
<process>	HPOM process to be traced or debugged.

   Note: Spaces are not allowed between entries in the lists for each process or area.

   The following examples illustrate how to enable tracing for the message/action flow and initialization and debug. Generate trace output only for `opcmsga` and `opcacta`. Enable debug output only for `opcmsga`.

   Management Server Configuration Commands

   ```
   ovconfchg -ovrg server -ns opc -set OPC_TRACE TRUE
   ovconfchg -ovrg server -ns opc -set OP_TRACE_AREA MSG,ACTN,INIT,DEBUG
   ovconfchg -ovrg server -ns opc -set OPC_TRC_PROCS opcacta,opcmsga
   ovconfchg -ovrg server -ns opc -set OPCDBG_PROCS opcmsga
   ```

   If the granularity of the above tracing options is not sufficient, use the variable `OPC_RESTRICT_TO_PROCS` to enable tracing for a particular area of a HPOM process.

3. To receive verbose trace information output, enter the following command:

   ```
   ovconfchg -ovrg server -ns opc -set OPC_TRACE_TRUNC FALSE
   ```

   By default, `OPC_TRACE_TRUNC TRUE` is enabled.

   For more information on tracing configuration, see "Examples of Tracing" on page 205.

Examples of Tracing

This section contains some examples to show how tracing can be activated for different areas and processes.

Enter the appropriate command:

- **Default**

  Collect trace information for the trace areas MSG (message flow) and ACTN (actions).

  ```
  ovconfchg -ovrg server -ns opc -set OPC_TRACE TRUE
  ```

- **Tracing for Heartbeat Polling and Message Flow**

  Collect trace information for the trace area ALIVE (agent-alive check).

  ```
  ovconfchg -ovrg server -ns opc -set OPC_TRACE TRUE -set OPC_TRACE_AREA ALIVE
  ```

- **Tracing for Specific Areas of Specific Processes**

  Collect trace information for the trace area API (application programming interface) of the Message Manager process opcmsgm.

  ```
  ovconfchg -ovrg server -ns opc -set OPC_TRACE TRUE -set OPC_TRACE_AREA API -set
  OPC_TRC_PROCS opcmsgm
  ```

- **Tracing and Debugging**

  - Collect trace information for all trace areas (except PERF), as well as debug information for all debug areas. Debug areas are to be used by HP Support Personnel only.

    ```
    ovconfchg -ovrg server -ns opc -set OPC_TRACE TRUE -set OPC_TRACE_AREA
    ALL,DEBUG
    ```

  - Collect trace information for all trace areas (except PERF) for the process ovoareqsdr (request sender), as well as debug information for all debug areas of the process ovoareqsdr (request sender).

    ```
    ovconfchg -ovrg server -ns opc -set OPC_TRACE TRUE -set OPC_TRACE_AREA
    ALL,DEBUG -set OPC_TRC_PROCS ovoareqsdr -set OPC_DBG_PROCS ovoareqsdr
    ```

- **Different Trace Areas for Different Processes**

  Restricting tracing to a specified process must specify the process in the tracing command.

  The areas to be traced are specified as usual.

  The first configuration entry enables tracing for the trace areas INIT (initialization) and INT (internal) of the control agent process (opcctla). The second configuration entry enables tracing for the trace areas MSG (message flow) and ACTN (actions) of the message agent process (opcmsga).

  ```
  ovconfchg -ovrg server -ns opc.opcctla -set OPC_TRACE TRUE -set OPC_TRACE_AREA
  INIT,INT

  ovconfchg -ovrg server -ns opc.opcmsga -set OPC_TRACE TRUE
  ```

Example of Tracing HPOM Processes

The following sample procedure provides an example of how to set up HP tracing on HPOM processes. The example makes the following configuration assumptions:

- The opcmsga and opcmsgm process running on a UNIX system must be traced.
- The ovtrccfg trace configuration client will be used to make configuration changes.
- The trace configuration file must be named: $OV_CONF/OVOTrace.tcf
- The ovtrcmon trace monitor client will be used to monitor the traces.
- The trace output must be written to a file named: $OV_LOG/OVOTrace.trc

To set up tracing on HPOM processes:

1. Identify the HPOM processes that you want to trace. (The following example uses the opcmsga and opcmsgm processes).
2. Create a trace configuration file named OvoTrace.tcf. Locate the file in the $OV_CONF directory.

   This sample trace configuration file enables tracing on the two HPOM applications: opcmsga and opcmsgm. The Sink is configured as a socket with the machine supnode1 as the target server. The components selected are the opc and eaagt. All the associated sub-components are selected except for the DEBUG sub-components. This would correspond to selecting All Areas except DEBUG. The tracing attributes are set to the Support defaults of Info, Warn, and Error for all, with the Verbose attribute added to each component/sub-component combination entry.

   Trace Configuration File $OV_CONF/OVOTrace.tcf

```
TCF Version 3.2
APP: "opcmsgm"
SINK: Socket "supnode1" "node=10.111.1.21;"
TRACE: "opc.actn" "Trace" Info Warn Error Developer Verbose
TRACE: "opc.agtid" "Trace" Info Warn Error Developer Verbose
TRACE: "opc.alive" "Trace" Info Warn Error Developer Verbose
TRACE: "opc.api" "Trace" Info Warn Error Developer Verbose
TRACE: "opc.audit" "Trace" Info Warn Error Developer Verbose
TRACE: "opc.db" "Trace" Info Warn Error Developer Verbose
TRACE: "opc.dist" "Trace" Info Warn Error Developer Verbose
TRACE: "opc.fct" "Trace" Info Warn Error Developer Verbose
TRACE: "opc.gui" "Trace" Info Warn Error Developer Verbose
TRACE: "opc.init" "Trace" Info Warn Error Developer Verbose
TRACE: "opc.inst" "Trace" Info Warn Error Developer Verbose
TRACE: "opc.int" "Trace" Info Warn Error Developer Verbose
TRACE: "opc.lic" "Trace" Info Warn Error Developer Verbose
TRACE: "opc.mem" "Trace" Info Warn Error Developer Verbose
TRACE: "opc.memerr" "Trace" Info Warn Error Developer Verbose
TRACE: "opc.misc" "Trace" Info Warn Error Developer Verbose
TRACE: "opc.mon" "Trace" Info Warn Error Developer Verbose
TRACE: "opc.msg" "Trace" Info Warn Error Developer Verbose
TRACE: "opc.name" "Trace" Info Warn Error Developer Verbose
TRACE: "opc.nls" "Trace" Info Warn Error Developer Verbose
```

```
TRACE: "opc.ntprf" "Trace" Info Warn Error Developer Verbose
TRACE: "opc.ocomm" "Trace" Info Warn Error Developer Verbose
TRACE: "opc.pdh" "Trace" Info Warn Error Developer Verbose
TRACE: "opc.perf" "Trace" Info Warn Error Developer Verbose
TRACE: "opc.pstate" "Trace" Info Warn Error Developer Verbose
TRACE: "opc.sec" "Trace" Info Warn Error Developer Verbose
TRACE: "opc.srvc" "Trace" Info Warn Error Developer Verbose
TRACE: "opc.wmi" "Trace" Info Warn Error Developer Verbose
APP: "opcmsga"
SINK: Socket "supnode1" "node=10.111.1.21;"
TRACE: "eaagt.actn" "Trace" Info Warn Error Developer Verbose
TRACE: "eaagt.agtid" "Trace" Info Warn Error Developer Verbose
TRACE: "eaagt.alive" "Trace" Info Warn Error Developer Verbose
TRACE: "eaagt.api" "Trace" Info Warn Error Developer Verbose
TRACE: "eaagt.audit" "Trace" Info Warn Error Developer Verbose
TRACE: "eaagt.db" "Trace" Info Warn Error Developer Verbose
TRACE: "eaagt.dist" "Trace" Info Warn Error Developer Verbose
TRACE: "eaagt.fct" "Trace" Info Warn Error Developer Verbose
TRACE: "eaagt.gui" "Trace" Info Warn Error Developer Verbose
TRACE: "eaagt.init" "Trace" Info Warn Error Developer Verbose
TRACE: "eaagt.inst" "Trace" Info Warn Error Developer Verbose
TRACE: "eaagt.int" "Trace" Info Warn Error Developer Verbose
TRACE: "eaagt.lic" "Trace" Info Warn Error Developer Verbose
TRACE: "eaagt.mem" "Trace" Info Warn Error Developer Verbose
TRACE: "eaagt.memerr" "Trace" Info Warn Error Developer Verbose
TRACE: "eaagt.misc" "Trace" Info Warn Error Developer Verbose
TRACE: "eaagt.mon" "Trace" Info Warn Error Developer Verbose
TRACE: "eaagt.msg" "Trace" Info Warn Error Developer Verbose
TRACE: "eaagt.name" "Trace" Info Warn Error Developer Verbose
TRACE: "eaagt.nls" "Trace" Info Warn Error Developer Verbose
TRACE: "eaagt.ntprf" "Trace" Info Warn Error Developer Verbose
TRACE: "eaagt.ocomm" "Trace" Info Warn Error Developer Verbose
TRACE: "eaagt.pdh" "Trace" Info Warn Error Developer Verbose
TRACE: "eaagt.perf" "Trace" Info Warn Error Developer Verbose
TRACE: "eaagt.pstate" "Trace" Info Warn Error Developer Verbose
TRACE: "eaagt.sec" "Trace" Info Warn Error Developer Verbose
TRACE: "eaagt.srvc" "Trace" Info Warn Error Developer Verbose
TRACE: "eaagt.wmi" "Trace" Info Warn Error Developer Verbose
```

If you have access to a Windows system with the `TraceMon` tool installed, it can be used to connect to the remote trace server to identify the application, component, and category names and to view the attributes.

3. Verify that the trace server is running on the system by executing the command:

   ```
   ps -ef | grep ovtrcd
   ```

   If the process is running, the information returned should be of the following form:

```
root 18750 1 0 Mar 5 ?0:00 /opt/OV/bin/ovtrcd
```

4. Verify that the applications being traced `opcmsgm`, are running on the system.

   To verify a process is running, execute commands of the following form:

   ```
   ovc -status opcmsga opcmsgm
   ```

   The information returned should be of the following form:

   ```
   opcmsgm  OMU  Message Manager  SERVER,OPC   (14038)  Running
   opcmsga  OMU  Message Agent    AGENT,EA     (5380)   Running
   ```

5. Use the `ovtrccfg` configuration client to set the tracing configuration, using the command:

   ```
   $OV_BIN/ovtrccfg -server supnode1 $OV_CONF/OvoTrace.tcf
   ```

6. Use the `ovtrcmon` monitor client to monitor the trace messages generated from the `opcmsga` and `opcmsgm` applications. To monitor the trace server running on the `supnode1` system and output the trace messages in binary format to the `$OV_LOG/OvoTrace.trc file`, enter the command:

   ```
   $OV_BIN/ovtrcmon -server supnode1 -tofile $OV_LOG/OvoTrace.trc
   ```

7. Provided that the processes to be traced are running (`opcmsga` and `opcmsgm` in our example), they should now be generating trace messages. Once enough trace information has been captured, stop the tracing. To Stop tracing, enter the command:

   ```
   $OV_BIN/ovtrccfg off
   ```

8. View the trace output using the `ovtrcmon` monitor client. The trace output can be read from the binary trace file created using the `ovtrcmon -fromfile` option. This option reads in a binary trace file and converts it to text. The converted trace messages can be sent directly to standard out or can be redirected to trace text file.

   To convert the binary trace file to text and send the output to standard out, enter the following command:

   ```
   $OV_BIN/ovtrcmon -fromfile $OV_LOG/OvoTrace.trc
   ```

   To redirect the converted trace messages to a text file, enter the following command:

   ```
   $OV_BIN/ovtrcmon -fromfile $OV_LOG/OvoTrace.trc > /tmp/trc.text
   ```

   The binary `$OV_LOG/OvoTrace.trc` can be viewed from within the TraceMon Windows tool, where additional filtering can be done.

9. If analysis of the trace output is inconclusive, additional tracing can be done to capture more trace information. If needed, the trace configuration file can be modified to include or remove applications, components, categories or attributes.

Configuring HTTPS-based Communication

HP applications may be customized for installation using configuration parameters. The communication broker configuration parameters are contained in the `bbc.ini` file located at:

<OVDataDir>`/conf/confpar/bbc.ini`

The parameters used for communication are described in the `bbc.ini(4)` file, which is described in the HP Operations agent documentation.

For more information about configuring HTTPS-based communication, see the *HPOM Concepts Guide*.

Synchronization of Configuration Data from One HPOM Server to Another

To use HTTPS-based communication for the transfer, make sure the source HPOM management server is set up as an action-allowed manager on the target HPOM server.

To allow synchronization of configuration data from one HPOM server to another by using HTTPS-based communication, follow these steps:

1. Create the appropriate configuration download information by running the `opccfgdwnld` CLI on the source HPOM server.
2. Run the following commands on the source HPOM server:

```
#!/usr/bin/sh
PATH=$PATH:/opt/OV/bin/OpC/install
tar cvf - /var/opt/OV/share/tmp/OpC_appl/cfgdwn | gzip > /tmp/cfgdwn.tar.gz
opcdeploy -deploy -file /tmp/cfgdwn.tar.gz -node mgmtsv2 -targetdir /tmp -trd
absolute
opcdeploy -cmd "rm -rf /var/opt/OV/share/tmp/OpC_appl/cfgdwn" -node mgmtsv2
opcdeploy -cmd "gunzip < /tmp/cfgdwn.tar.gz| tar xvf - 2>&1"  -node mgmtsv2
```

3. Upload the configuration on the target HPOM server by running the `opccfgupld` CLI at a convenient time (for example, the planned maintenance window of the targeted HPOM server).

Chapter 4: HPOM Software Distribution to Managed Nodes

In this Chapter

This chapter describes how to distribute the HPOM configuration to managed nodes. The information in this chapter covers the following topics:

HPOM Agent-Configuration Distribution

After customizing the configuration and assigning policies to managed nodes, distribute the agent configuration to the managed nodes again using the `opcragt` command. If no configuration change has been made since the last configuration distribution, no new distribution is triggered unless you use the `-force` option. For more information about command options and parameters, see the *opcragt(1m)* manual page.

Instrumentation Distribution

This section contains the general recommendations before distributing commonly used instrumentation data to the managed nodes. You can call this data as automatic actions, operator-initiated actions, or scheduled actions. It can also be used by the monitoring agent and log file encapsulator.

Before You Distribute Instrumentation Data

Before you distribute instrumentation data to the managed nodes, review the following distribution requirements and tips.

Distribution Requirements

HPOM distributes instrumentation data only if one of the following is true:

- Deployment status:

  Instrumentation files are available on the management server but are not yet deployed on the managed node.

- Available versions:

  Instrumentation files available on the management server are newer than those already deployed to the managed node.

Distribution Tips for All Systems

To reduce network traffic and speed up distribution, follow these guidelines:

- Commonly used binaries:

  Put only commonly used binaries to the instrumentation data location on the HPOM management server. Choose the appropriate location considering the criteria provided with your chosen distribution method. For more information, see "Distribution Methods " on page 213.

- Customized binaries:

  If you need a certain binary to be present only on specific systems, place this binary at an appropriate location under the category you have created for this purpose (see "Before You Distribute Instrumentation Data" on page 218 for more information). For description of categories and the distribution method based on them, see "Category-based Distribution of Instrumentation" on page 215.

- Distribution process:

  If too many distribution requests are to be proceeded by the distribution process `opcbbcdist`, the other HPOM services (for example, the message manager) can be slowed down. By default, `opcbbcdist` handles 10 requests in parallel and the number of threads can be controlled using the `OPC_MAX_DIST_REQS` configuration setting.

  To avoid performance problems, do the following:

- Do not configure all managed nodes at one time:

  Minimize the number of managed nodes getting new configuration data at the same time:

  - Distribute configuration to only a few nodes at a time by using the `opcragt` command.
  - Set a low number for maximum distribution requests by using the `ovconfchg` command as illustrated in the following example:

    ```
    /opt/OV/bin/ovconfchg -ovrg server -ns opc -set OPC_MAX_DIST_REQS 3
    ```

Note: If you configure a low number for maximum distribution requests, but an immediate deployment to a managed node is required while a large number of requests is being handled,

you can use `opcragt -distrib -highprio <node>`. For usage details, see *opcragt(1m)* manual page.

- Reduce the process priority of `opcbbcdist`:

  Use the `renice(1)` command to reduce the process priority of `opcbbcdist` on the management server.

- Use category-based distribution method or selective distribution feature of `opcbbcdist`:

  Prevent distribution of the particular configuration files which are not needed on a specific node by choosing the category-based distribution method or the Selective Distribution feature of `opcbbcdist`. See "Category-based Distribution of Instrumentation" on page 215 for more information about categories. For details on Selective Distribution Feature, see "Selective Distribution to Managed Nodes" on page 224.

- Distribution directory:

  If you want to stop the distribution of configuration, scripts, or programs (for example, if the configuration is invalid), clean the `distrib` directory:

  ```
  # /var/opt/OV/share/tmp/OpC/distrib
  ```

  Clean the `distrib` directory only in an emergency and only after the HP Operations management server processes have been stopped.

Distribution Methods

HPOM provides several different ways of distributing data to the managed nodes. The most appropriate distribution method depends on not only on the scope of the data you want to distribute but also on the selection of managed nodes to which you want to distribute the data. For example, you can distribute the complete instrumentation to a number of managed nodes or selected parts of the instrumentation to one, particular managed node.

Distributing All Instrumentation

To distribute all instrumentation data to all specified managed nodes, choose one of the following methods. Note that it is recommended to distribute instrumentation by category:

- Distribution based on instrumentation category:

  For more information about distributing instrumentation based on categories, see the following sections:

  a. "Category-based Distribution of Instrumentation" on page 215.

  b. "Before You Distribute Instrumentation Data" on page 218.

- Distribute from instrumentation directories:

  For more information about distributing instrumentation directly from the directories in which the instrumentation is located, see the following section:

  a. "Distribution of Instrumentation to Managed Nodes " on page 221.

Distributing Selected Instrumentation

To distribute only specific files to a particular managed node, choose one of the following methods. Note that it is recommended to distribute instrumentation by category:

- Distribute instrumentation to managed nodes based on categories. For more information, see the following sections:
  - "Category-based Distribution of Instrumentation" on page 215.
  - "Before You Distribute Instrumentation Data" on page 218.
- Distribute selected instrumentation files to specific managed nodes. For more information, see the following section:
  - "Selective Distribution to Managed Nodes" on page 224.

Simulating Distribution

During deployment, it could be difficult to predict which files are deployed to a node. In such cases, it is useful to simulate the distribution using the `-simulate` option of the command-line utility `opcragt` before starting the actual deployment.

The `-simulate` option could be used together with all other deployment-related options except `-subagent`. The output is a list of all files (policies and instrumentation) that would be actually deployed to a node, including the full paths to these files.

Deployment simulation feature can be used only on one node at the time.

Error Logging

When the instrumentation is distributed to a managed node on which the agent is not installed, the error is printed in the `/var/opt/OV/log/System.txt` file.

This error message is logged only once. However, if you want to disable logging of these errors, enter the following:

```
ovconfchg -ovrg server -ns opc -set OPC_DIST_OMIT_ERROR_AGT_NOT_INST true
```

Category-based Distribution of Instrumentation

This section describes the distribution of instrumentation to managed nodes based on **categories**. An instrumentation category is a concept used to group related instrumentation files in logical units.

The possibility to group the instrumentation files into categories simplifies their distribution to the particular managed nodes. Customized scripts and programs can be grouped in a category, for example, `Custom`, which is then assigned to the specific managed nodes. Upon distribution, these scripts and programs are deployed only to the managed nodes to which the category is assigned to.

It is possible to deploy only the specified files to managed nodes because of the multi-level directory structure inside the categories. Each category can contain the specific instrumentation files in its directory substructure, as described in "Instrumentation Data Directory Structure" on page 215.

Category information is stored in the HPOM database, and can be managed simultaneously in the file system and on the database level by means of the `opcinstrumcfg` command-line utility. For more information about command options and parameters, see the *opcinstrumcfg(1m)* manual page.

For the information about the category-related database tables, see the *HPOM Reporting and Database Schema*.

Note: Categories are used also for versioning instrumentation files (scripts and binaries), because they cannot have version numbers assigned in HPOM.

For information about managing multiple versions of the instrumentation data, see "Managing Multiple Versions of HPOM Configuration on Managed Nodes" on page 147.

Instrumentation Data Directory Structure

The instrumentation data is organized differently according to whether it is deployed on the HPOM management server or on the managed node. For more information about the locations where HPOM deploys instrumentation data, see the following sections:

- "Instrumentation on the HPOM Management Server" on page 215
- "Instrumentation Data on the HPOM Managed Nodes" on page 217

Instrumentation on the HPOM Management Server

The directory for executable files on the HP Operations management server is located in:

```
/var/opt/OV/share/databases/OpC/mgd_node/
```

When you create a category for instrumentation files, HPOM automatically creates a multi-level subdirectory structure where the instrumentation files, organized by category, are configured for the distribution.

Note: HPOM deploys all instrumentation data located in `/var/opt/OV/share/databases/OpC/mgd_node/`, including the contents of the `monitor`, `actions`, and `cmds` directories. However, if there are files in `monitor`, `actions`, and `cmds` with the same names as files within the created categories, the files organized in categories are distributed in preference to the files in the `monitor`, `actions`, and `cmds` directories.

The subdirectory structure under the `instrumentation` directory can be one of the following:

- `$InstrumDir/<category>/<OS_family>/<OS_type>/<cpu_type>/<OS_version>`
- `$InstrumDir/<category>/<OS_family>/<OS_type>/<OS_version>/<cpu_type>`

Caution: Only one subdirectory structure may be used. Nevertheless, if both subdirectory structures are used at the same time, the following one is ignored:

`$InstrumDir/<category>/<OS_family>/<OS_type>/<OS_version>/<cpu_type>`

The following list explains the selectors used in the instrumentation directories structure:

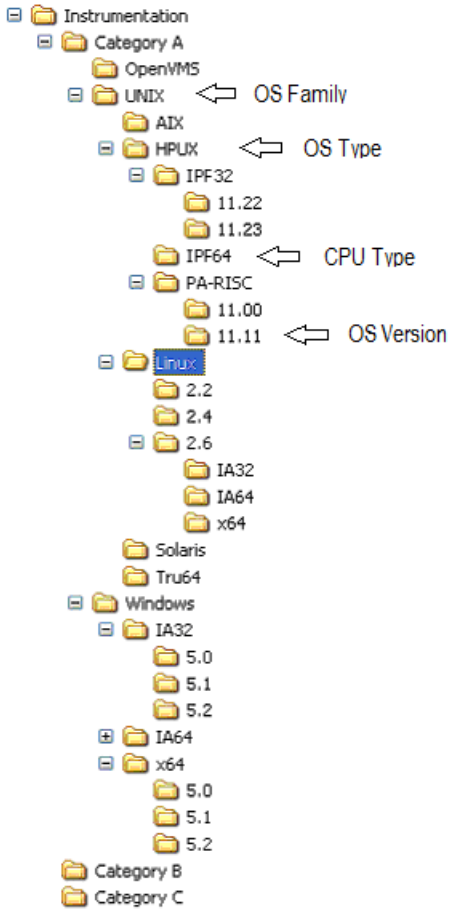
`$InstrumDir`	`/var/opt/OV/share/databases/OpC/mgd_node/instrumentation`
<OS_family>	Unix, Windows
<OS_type>	Windows, Linux, HP-UX, Solaris, AIX, and OpenVMS Both Windows and OpenVMS combine the OS Family and OS type into a single directory level. For example, `$InstrumDir/<category>/Windows/X86/...` and not `$InstrumDir/<category>/Windows/Windows/...`
<cpu_type>	IPF32, IPF64, x64, x86, PA-RISC, SPARC, PowerPC, and Alpha
<OS_version>	All agent OS versions supported by HPOM 9.xx. For more information, see the *HPOM Software Release Notes*. The following mapping is used for the OS versions in the Microsoft Windows family: • Windows 2000 = 5.0 • Windows XP = 5.1 • Windows 2003 = 5.2 • Windows Vista = 6.0 • Windows 2008 = 6.0

- Windows 2008r2 = 6.1
- Windows 7 = 6.1
- Windows Server 2012 = 6.2
- Windows 8 = 6.2

Note: The OS version directory can reside under the *<OS_type>* or *<cpu_type>* directory. However, if there is no specific instrumentation data for a certain agent OS version, the corresponding subdirectory is not created in the file system.

Figure 5 shows the instrumentation directory structure on the HPOM management server.

Figure 5: Instrumentation Directory Structure on the HPOM Server



Instrumentation Data on the HPOM Managed Nodes

On HPOM managed nodes, all deployed instrumentation data (category-based instrumentation, as well as the files for monitors, actions, and commands) are located in the following directory:

```
/var/opt/OV/bin/instrumentation
```

Before You Distribute Instrumentation Data

Before you start the distribution of instrumentation, consider the following important points:

- Distributing *all* instrumentation:

  If you want to deploy all the files from the `instrumentation` directory on the server to all specified managed nodes, create a subdirectory named `default` inside the `$InstrumDir/instrumentation` directory, move all the files to be distributed into this new directory, and start the distribution process.

- Distributing instrumentation to specific nodes:

  If you want to deploy instrumentation files to particular managed nodes, create and assign categories only to these target nodes, then start the distribution process.

- Distributing *specific* instrumentation:

  If you want to deploy user-specified instrumentation files to the managed nodes, make sure you have placed the files you want to distribute in the appropriate location under the category you have created for this purpose. For example, if you want to deploy only the instrumentation related to IPF32 11.23 under a category `Custom`, create the new category, which creates the following subdirectory structure:

  ```
  $InstrumDir/Custom/UNIX/HP-UX/IPF32/11.23/
  ```

  Make sure the category Custom is assigned to the appropriate managed nodes, and then start the distribution process.

Distributing Instrumentation by using Categories

To distribute instrumentation data to managed nodes using *custom* categories, perform the following steps in the indicated order. For more information on how to perform each individual step, see the sections that follow:

1. Create the `Custom` category. For more information, see "Creating Instrumentation Categories" on page 219.
2. Place the instrumentation files in the appropriate location. For more information, see "Locating Instrumentation Data" on page 219.
3. Assign the category `Custom` to the appropriate managed nodes and to the appropriate policies. For more information, see "Assigning Instrumentation Categories" on page 220.
4. Start the distribution process using the `opcragt` command-line utility with the `-instrum` option. For more information about the distribution process, see "Deploying Instrumentation Data " on page 221.

Note: If you want to check which files will be distributed to a node before you actually start the distribution, you can use a deployment simulation feature provided with the `opcragt` command-line utility. For more information, see "Simulating Distribution" on page 214.

Creating Instrumentation Categories

To create categories for the custom instrumentation you want to distribute, you can choose any of the following methods:

- `opcinstrumcfg` utility:

  To use the `opcinstrumcfg` utility to create a new instrumentation category, enter the following command:

  ```
  # opcinstrumcfg -add <categoryA>,<categoryB>
  ```

  Note that the items in a list of categories must be separated by a comma since the category names can contain blank characters.

  The `opcinstrumcfg` utility enables you to manage categories both on a file system and on a database level. For more information about command options and parameters, see the opcinstrumcfg(1m) manual page.

- `opcpolicy` utility:

  To use the `opcpolicy` utility to create a new instrumentation category, enter the following command:

  ```
  # opcpolicy -add_cat cat_list=<categoryA>,<categoryB>create=[yes|no]
  ```

  Note: If you specify `create=no` (the default is `yes`), no subdirectory structure is created under these new categories in the file system.

  `opcpolicy` is a symbolic link to the `opctempl` command line utility, and is used for managing policies. For more information about command options and parameters, see the opcpolicy(1m) manual page. For more detailed information about policies including how to create them, see the *HPOM Concepts Guide*.

- `opccfgupld` and `opccfgdwn` utilities:

  During the upload or download of the configuration data, the category assignments to policies are also added to the database, since they are regular policy header attributes. For more information about uploading and downloading the configuration data, see the *opccfgupld(1m)* and *opccfgdwn (1m)* manual pages.

Locating Instrumentation Data

To make sure the deployment process completes successfully, you must place the instrumentation data in the correct location within the instrumentation subdirectory structure. HPOM expects to find

instrumentation data in the following location:

```
$InstrumDir/Custom/<OS_family>/<OS_type>/<cpu_type>/<OS_version>
```

For example, you place custom instrumentation data for the HP-UX 11.31 operating system version in the following location:

```
$InstrumDir/Custom/Unix/HP-UX/IPF32/11.31/
```

For more information about where HPOM expects to find instrumentation data, see the following sections:

- "Before You Distribute Instrumentation Data" on page 218
- "Instrumentation Data Directory Structure" on page 215

Assigning Instrumentation Categories

Depending on whether you want to assign an instrumentation category to a managed node, a policy, or a policy group, choose one of the following methods:

- *Assigning an instrumentation category to a managed node:*

  To assign an instrumentation category to a managed node, use the `opcnode` command:

  ```
  # opcnode -assign_cat node_list=<node_list> cat_list=<category_list>
  ```

  For more information about command options, see the *opcnode(1m)* manual page.

- *Assigning an instrumentation category to a node group:*

  To assign an instrumentation category to a node group, use the `opcnode` command:

  ```
  # opcnode -assign_cat group_name=<group_name> cat_list=<category_list>
  ```

  For more information about command options, see the *opcnode(1m)* manual page.

- Assigning an instrumentation category to a policy:

  To assign an instrumentation category to a policy, use the `opcpolicy` command:

  ```
  # opcpolicy -update policy=<policy_name> type=<policy_type> [version=<policy_
  version>] add_cats=<categoryA>,<categoryB>
  ```

  Note: Use the *version* option to assign an instrumentation category to a specific policy version. Otherwise, the instrumentation category is assigned to all versions of this policy.

  When assigning an instrumentation category with the `opcpolicy` command, note the following prerequisites:

  - Instrumentation categories must be assigned to a particular policy (for example, `policyX`).
  - The policy (`policyX`) must be assigned to the managed nodes to which you want to distribute the instrumentation data.

For more information about policy management, see "HPOM Policies" on page 71. For more detailed information, see the *HPOM Concepts Guide*.

- Assigning an instrumentation category to a policy group:

  To assign an instrumentation category to a policy group, use the `opcpolicy` command:

  ```
  # opcpolicy -assign_cat_to_group pol_group=<policy_group_name> cat_list=<comma_separated_categories>
  ```

  For more information about command options, see the *opcpolicy(1m)* manual page.

Deploying Instrumentation Data

To start the deployment of instrumentation data to managed nodes using the category-based distribution method, run one of the following commands on the HPOM management server:

- Deploy all instrumentation data, including the contents of the `monitor`, `actions`, and `cmds` directories using the following command and parameters:

  ```
  # /opt/OV/bin/OpC/opcragt -distrib -instrum <node_name>
  ```

<node_name>	Name of the node to which you want to deploy the instrumentation data.

  If any files in the `monitor`, `actions`, and `cmds` directories have identical names to files in the `instrumentation` directory, HPOM deploys the files from the `instrumentation` directory.

- Deploy all instrumentation data using the following command *only* if the policies are updated with the required category assignments:

  ```
  # /opt/OV/bin/OpC/opcragt -distrib -policies
  ```

  The `opcragt` command with the `-policies` option deploys policies and subsequently all categories assigned to these policies are added to the database. For more information about command options and parameters, see the *opcragt(1m)* manual page.

Distribution of Instrumentation to Managed Nodes

This section explains how to distribute commonly used instrumentation data from monitor, commands, and actions directories to the managed nodes.

Before You Distribute Instrumentation Data

Before you distribute instrumentation data from monitor, actions, and commands to the managed nodes, beside the general recommendations review also the following distribution requirements and tips:

- Customized scripts:

  Specify the full path name of the customized script in the HPOM configuration. Or make sure the file is available through the *$PATH* settings of the executing user on the managed node.

  For example, a customized script to determine running processes might look like one the following:

  a. `/name/opc_op/scripts/my_ps`

  b. `my_ps`

  You can call this script as an application from the Java GUI or as a broadcast command.

- Customized binaries:

  HPOM compresses the monitors, actions, and command binaries. If a file with a `.Z` extension already exists, do not put files into the following directory:

  `/var/opt/OV/share/databases/OpC/mgd_node/customer/<arch>/{monitor|actions|cmds}`

  <arch> Architecture of the monitors, actions, and commands, for example: `hp/alpha/tru64` or `sun/sparc/solaris7`.

  When distributing scripts to managed nodes on UNIX systems, follow these guidelines:

- Mixed clusters:

  You must install the scripts and programs for monitors, actions, and commands only once for each architecture type. For each architectural type, select one cluster node.

- File names:

  The file names of the monitors, actions, and commands binaries must not be longer than 14 characters (including the `.Z` extension if the binary is compressed). This limitation is set to ensure smooth processing on nodes running with short file names.

Instrumentation Data Distribution

You can distribute instrumentation data by using the `opcragt` command line interface. Instrumentation files are distributed only if they are not already installed on the managed node, or when a newer version is available on the management server.

Note: To update only the changes in the configuration, do not use the `-force` option. The `-force` option (re-)distributes all files, which can cause an increase in network load.

For information about the directories on the management server and the managed node, see "Instrumentation Data Locations" on page 223.

The binaries are located in the temporary directories only during the distribution phase. When distribution is completed, the local HPOM action and monitor agents are stopped, the binaries are moved or copied to their final destination, and the HPOM action and monitor agents are restarted.

The HPOM action agent and monitor agent append directories to the *$PATH* setting of the executing user.

Note: If you want to check which files will be distributed to a node before you actually start the distribution, you can use a deployment simulation feature provided with the `opcragt` command-line utility. For more information, see "Simulating Distribution" on page 214.

Instrumentation Data Locations

HPOM organizes instrumentation data differently according to whether it resides on the management server or the managed node. For example, on the management server, instrumentation data is split into vendor-related and customer-related areas. For more detailed information about the location of instrumentation data see the following sections:

- "Instrumentation on the HPOM Management Server" on page 223
- "Instrumentation on the HPOM Managed Node" on page 223

Instrumentation on the HPOM Management Server

Instrumentation files are located in the following two directories on the HP Operations management server:

- `/var/opt/OV/share/databases/OpC/mgd_node/vendor/<arch>[/<comm>]/actions|cmds|monitor`
- `/var/opt/OV/share/databases/OpC/mgd_node/customer/<arch>[/<comm>]/actions|cmds|monitor`

The replaceable elements *<arch>* (hardware architecture) and *<comm>* (communication type) combine to define the directory specific to the operating system and, if desired, the communication type of the node to which you want to deploy the instrumentation data files.

The vendor-specific files contained within directory structure `/var/opt/OV/share/databases/OpC/mgd_node/vendor` are used for the default configuration of HPOM and are always distributed. The files contained in the customer tree are required only if policies are assigned and distributed.

Note: If identical files for `actions`, `cmds`, and `monitor` exist in both the `customer` and `vendor` directories, use the `customer` files.

Instrumentation on the HPOM Managed Node

On HPOM managed nodes, all instrumentation data (category-based instrumentation, and the `actions`, `cmds`, and `monitor` files) is located in the following directory:

`/var/opt/OV/bin/instrumentation`

Selective Distribution to Managed Nodes

This section describes the selective distribution feature, which you start with the `opcbbcdist` command and configure with the `seldist` configuration file.

`opcbbcdist` usually distributes all the files to managed nodes from two sets of directories corresponding to the selected managed node type, for example HP-UX or Windows. For their location, see "Instrumentation Data Locations" on page 223.

During the default distribution process, HPOM deploys all instrumentation data including some files that might not be needed on a specific node. The problem of deploying unnecessary files is especially noticeable with the HP Operations Smart Plug-ins (SPIs). The SPI binaries can be very large and when distributed to all target nodes, may occupy a significant amount of network bandwidth during distribution and large amounts of disk space on the managed nodes.

The Selective Distribution functionality gives you greater flexibility in distributing files from the HP Operations management server. You can prevent distribution of a user-selected set of files and binaries, for example, files belonging to a SPI, from `actions`, `cmds`, and `monitor` directories to specific nodes that do not belong to the node group associated with the SPI.

The file `seldist` enables you to configure a list of files and target node groups that you have selected for deployment. For more information about the `seldist` configuration file, see "seldist Configuration File" on page 225.

The advantages of the selective-distribution feature include the following:

- Reduced disk-space utilization on managed nodes
- Reduced network traffic during configuration-file distribution

If selective distribution is *not* enabled, HPOM performs a standard distribution from monitors, actions and commands. If you want to avoid distributing *all* instrumentation data, use the category-based distribution method since it also allows you to distribute specified user-selected files to a particular managed node.

See "Category-based Distribution of Instrumentation" on page 215 for more information. See also "Distribution Methods " on page 213 to learn about the available distribution methods.

Selective Distribution Startup

On starting configuration file distribution from the command line, `opcbbcdist` checks the selective distribution configuration. When the distribution process of `actions`, `commands` or `monitors` is started, Selective Distribution in accordance with the requirements of the `seldist` file is started.

On distribution, each file from the customer actions, commands, and monitors directories is compared against each file name prefix in the `seldist` file. If it does not match any prefix, it is distributed to all agents of the respective platform.

If it matches one or more entries, it is only distributed to the agents of the corresponding node group(s). For example, an empty `seldist` file would result in all files being distributed to all nodes.

In a flexible-management environment, you must *manually* ensure synchronization of the `seldist` files on all of your HP Operations management servers.

Most of the files installed by the Database SPI have the `dbspi` prefix. SAP SPI files have an `r3` prefix. For example, a SAP SPI binary would be named `r3perfmon`.

In addition to the preconfigured SPI-related files, you can also add your own files and file prefixes together with a node-group name. This is most useful if you have your own policies and accompanying scripts that only need to be distributed to a subset of the nodes. For more information, see the section "Configuring Custom Selective Distribution" on page 230.

seldist Configuration File

A `seldist` configuration file is provided in which node group names together with file name prefixes and files are listed. This file is either read by `opcbbcdist` on startup or by the `opcseldist` utility for selective-distribution process. For more information about the `opcseldist` utility, including usage examples and command-line options, see "opcseldist Utility" on page 228 or the *opcseldist(1m)* manual page.

Selective distribution is automatically enabled if the `seldist` file exists in the following directory: `/etc/opt/OV/share/conf/OpC/mgmt_sv/`

When the distribution of actions, commands, or monitors starts, the selective-distribution process uses the contents of the `seldist` file to distribute the instrumentation data.

The list of files in `seldist` refers only to files within the following directory tree:

`/var/opt/OV/share/databases/OpC/mgd_node/customer/<arch>[/<comm>]`

The `seldist` configuration file lists, for each SPI, the target node group and a list of files and file prefixes that belong to this SPI. To include a managed node in the selective-distribution process, you must add the managed node to the node group specified in the `seldist` file.

All files that are not listed in the `seldist` file are also distributed to all nodes. In this way, the distribution process is backwards compatible with the standard distribution of `actions`, `cmds`, and `monitor` as only certain “known” files are blocked from distribution to nodes that do not belong to a specific group of nodes.

Activating the Selective-Distribution Process

The configuration file, `seldist.tmpl`, contains information regarding the file-name prefixes for all currently known SPIs including suggested node-group names. To use this selective-distribution configuration file, make a copy of the default selective-distribution template (`seldist.tmpl`) and place it in the same directory. The name the new copy must be "`seldist`".

For more information, see the section "Enabling Selective Distribution" on page 228. The following example shows an extract from a sample `seldist.tmpl` file.

Selective-Distribution Configuration File

```
# This is the specification file for Selective Distribution.
# It is delivered as:
#/etc/opt/OV/share/conf/OpC/mgmt_sv/seldist.tmpl.
# Before it can be used, the file has to be copied to:
# /etc/opt/OV/share/conf/OpC/mgmt_sv/seldist and edited there.

# Database SPI
#
DBSPI dbspi                             # general prefix for most files
DBSPI ntwdblib.dll                      # used for MS SQL on Windows
DBSPI sqlakw32.dll                      # used for MS SQL on Windows
DBSPI libopc_r.sl                       # used for Oracle 7.3.4 on HP-UX 11.00
# end of section Database SPI

# SPI for mySAP.com
#
sap r3                                  # general prefix for most files
sap sap_mode.sh
sap netperf.cmd                         # used for the NETPERF subagent
sap OvCor.dll                           # used for SAP on Windows
sap OvItoAgtAPI.dll                     # used for SAP on Windows
sap OvMFC.dll                           # used for SAP on Windows
sap OvR3Wrapper.dll                     # used for SAP on Windows
sap OvReadConfig.dll                    # used for SAP on Windows
sap OvSpiASER3.dll                      # used for SAP on Windows
sap librfc32.dll                        # used for SAP on Windows
# end of section SPI for mySAP.com

# PeopleSoft SPI
# This is partitioned into 4 node groups.
# The PS DB Server nodes need the files from the Oracle SPI as well.
#

PSAppServer psspi
PSBatchServer psspi
PSDBServer psspi
PSDBServer dbspi                        # used for the PS DB Server nodes
PSDBServer libopc_r.sl                  # used for Oracle 7.3.4 on HP-UX 11.00
```

```
PSWebServer psspi
# end of section PeopleSoft SPI
```

The syntax of the `seldist` file uses the following mandatory conventions:

- Comments:

  All text after a number sign (#) is treated as a comment and is not evaluated by the selective-distribution process.

- Active text:

  The selective-distribution process only evaluates the first two words are evaluated in any *enabled* (uncommented) lines, for example:

  ```
  DBSPI dbspi
  DBSPI ntwdblib.dll
  ```

  ```
  sap r3
  sap sap-mode.sh
  ```

  In these examples, the first word represents the name of the node-group targeted for the selective-distribution process, for example: `DBSPI` and `sap`. The second word represents either a file name prefix or an individual file. For example, `dbspi` and `r3` are file name prefixes, and `ntwdblib.dll` and `sap-mode.sh` are individual files.

  Note: All file names are treated as prefixes. For example, the file name `ntwdblib.dll` would also match the file `ntwdblib.dll.old`, which would be included in the selective distribution.

- Node-group name:

  The same node group can be specified several times. As a result, it is possible to specify multiple prefixes, file names, or both for the same node group.

  To specify a node group whose name includes a space, wrap the node name in double quotes, for example, `"node group 1" prefix1`. If a node-group name does not contain any spaces, you do not need to wrap the name in quotes.

  Node group names may be localized.

- File-name prefix:

  You can associate the same prefix with more than one node group to ensure the deployment of a common subset of files. For example, the PeopleSoft SPI ships certain DBSPI files that are required on a PeopleSoft database server:

  ```
  DBSPI dbspi
  PS_DB_Server dbspi
  ```

A file matching the dbspi prefix, for example, dbspicao, is distributed to a node only if that node belongs to either of the node groups DBSPI or "PS DB Server". Similarly, it is possible to specify prefixes that are subsets of each other.

Note: Any file names not included in the seldist file or that do not match any of the listed prefixes are distributed to *all* nodes, in the same way as they would be distributed to all nodes if the seldist functionality were not enabled.

opcseldist Utility

The opcseldist utility is a tool you can use to check the validity of seldist configuration files and send a reconfiguration request to opcbbcdist. The opcseldist utility has the following command line options:

-check *<filename>*	Checks the syntax of the file *<filename>*
-reconfig	Notifies opcbbcdist of changes to the seldist file and instructs it to use the modified content.

If the syntax of the configuration file is invalid, opcseldist displays a list of errors. If an invalid configuration file is used to start a selective distribution, the distribution manager evaluates the seldist file only until it encounters the first error; any configuration data after the error is ignored.

Enabling Selective Distribution

To enable selective distribution using the configuration file supplied by a Smart Plug-in, perform the following steps:

1. Create node groups for the nodes to which you want to distribute your instrumentation data, for example: actions, commands, and monitors.

   Most SPIs already come with default node groups for their specific configurations. However, you can use a different node-group name as long as you remember to modify the seldist file accordingly.
2. Make sure that all required nodes are included in the node groups that are targeted for selective distribution of the SPI-related files.
3. Change directory to the location of the selective-distribution configuration file, as follows:

   ```
   # cd /etc/opt/OV/share/conf/OpC/mgmt_sv
   ```
4. Make a copy of the seldist.tmpl file and rename the copied file to seldist, enter:

   ```
   # cp seldist.tmpl seldist
   ```

5. In the `seldist` file, locate the configuration section for the SPI that you want to configure and make the desired changes.

   Tip: To avoid problems during the selective-distribution process, check the configuration sections for all SPIs that you do *not* have installed and make sure that these sections are disabled.

6. Save the configuration file and check the syntax, as follows:

   ```
   # /opt/OV/bin/OpC/utils/opcseldist -check seldist
   ```

   Correct any possible syntax errors in the file.

7. Run the `opcseldist` utility to reconfigure `opcbbcdist`, as follows:

   ```
   # /opt/OV/bin/OpC/utils/opcseldist -reconfig
   ```

   The `opcbbcdist` process rereads the `seldist` configuration file and checks the database for node groups specified in the configuration file. Because of possibly unwanted side effects, `opcbbcdist` will report to both the message browser and the `System.txt` file node groups that display in the `seldist` file, but are not in the database.

   Note: The `opcbbcdist` process reads the `seldist` configuration file during each startup. However, if you edit the `seldist` file and want to make the changes effective instantly, run the `opcseldist` utility with the `-reconfig` option. For more information on the `opcseldist` utility, usage, and command-line options, see "opcseldist Utility" on page 228 or the *opcseldist(1m)* manual page.

8. Distribute the `actions`, `cmd`s, and `monitor` binaries using the `opcragt` command.
9. If you have already distributed SPI actions, commands, and monitor binaries to the managed nodes and you now want to *remove* unnecessary binaries from these nodes, run a selective-distribution with the `-purge` option.

   Note: If you have distributed instrumentation from several, different HPOM servers, the `-purge` option from one management server removes instrumentation distributed from another HP Operations server, too.

Disabling Selective Distribution

To disable selective distribution, performing the following steps:

1. Locate the active selective-distribution configuration file.

   Change to the following directory:

   ```
   # cd /etc/opt/OV/share/conf/OpC/mgmt_sv
   ```

2. Disable selective-distribution.

   Rename the active configuration file, `seldist`. For example, enter:

   ```
   # mv seldist seldist.old
   ```

3. Notify HPOM about the new selective-distribution configuration.

   Run the `opcseldist` command with the `-reconfig` parameter. Enter:

   ```
   # /opt/OV/bin/OpC/utils/opcseldist -reconfig
   ```

Configuring Custom Selective Distribution

The default `seldist` file currently contains configuration details for a selection of known SPIs. The configuration includes suggested names for node groups for the distribution of SPI-related files and binaries. You can configure a selective distribution of your own files and binaries placed in the `actions`, `cmds`, and `monitor` directories that you want to distribute to specified nodes or node groups, by creating a new configuration section in the `seldist` file.

To configure custom selective distribution, perform the following steps:

1. Update the `seldist` file.

   Modify the `seldist` configuration file by creating a new section for the instrumentation you want to distribute selectively. The new section should include the following information:

   - Node-group name:

     Name of the node group containing the managed nodes to which you want to selectively distribute instrumentation files.

   - File names and prefixes:

     Name (or prefix) of the file you want to include in the custom selective distribution.

     For more information about the syntax rules that are mandatory in the file you use to configure selective distribution, see "seldist Configuration File" on page 225.

2. Validate the changes you make to the `seldist` file.

   Run the `opcseldist` command with the `-check` parameter to make sure that the changes you made to the `seldist` file meet the syntax requirements and do not contain any errors. Enter:

   ```
   # /opt/OV/bin/OpC/utils/opcseldist -check seldist
   ```

3. Assign nodes to node groups, if necessary.

   Use the HPOM administrator's user interface to make sure that the managed nodes to which you want to distribute selected files are included in the node group specified in the new section of the `seldist` configuration file.

4. Notify HPOM of the changes made in the selective-distribution configuration file.

Run the `opcseldist` utility to force `opcbbcdist` to use the new configuration details. Enter:

```
# /opt/OV/bin/OpC/utils/opcseldist -reconfig
```

Chapter 5: Configuring HPOM to Use the IPv6

In This Chapter

This chapter instructs you on how to configure HPOM to communicate over the network by using the IPv6 protocol.

Before proceeding with the configuration of the IPv6 technology in your HPOM environment, get familiar with the following terms:

Internet Protocol	Internet protocol is a communication mechanism that provides a standard set of rules for sending and receiving data over the network.
IPv4	Internet protocol version 4.
IPv6	Internet protocol version 6.
Single-stack system	A system where only a single internet protocol is running. It uses either an IPv4 or an IPv6 address.
Dual-stack system	A system where both IPv6 and IPv4 protocols are running. It uses both IPv4 and IPv6 addresses.

For detailed descriptions, see the following sections:

- "IPv6 Support Specifics" on page 232
- "Configuring the IPv6 Protocol on the HP Operations Management Server" on page 234
- "IPv6 Protocol Limitations" on page 239

IPv6 Support Specifics

This section provides you with important information regarding the IPv6 protocol support in your HPOM environment. This information is presented as follows:

- "Required IP Communication Architectures" on page 233
- "HPOM Functionality Supported with IPv6" on page 234

Note: For the information about the IPv6 configuration prerequisites, how to enable the IPv6 protocol, configure it in the cluster environment, and how to determine whether it is present on your HP Operations management server, see "Configuring the IPv6 Protocol on the HP Operations

Management Server" on page 234.

Required IP Communication Architectures

To use HPOM with IPv6, the network architecture must be in accordance with the following requirements:

- **HP Operations management server**

  HP Operations management server fully supports both IPv4 and IPv6 protocols, as well as the dual-stack architecture. Because most of the computer networks still use the IPv4 protocol or the dual-stack architecture, HP Operations management server runs on the dual-stack system so that communication with the managed nodes with various communication stacks can be established.

  To configure server pooling in the HPOM environment where the IPv6 communication protocol is used, see *the High Availability Through Server Pooling White Paper*.

- **HP Operations Agent**

  The HP Operations Agent supports the following architectures for IP communication:

  - IPv4

    Caution: The 11.13 version of the HP Operations Agent provides the capability to communicate with IPv6-enabled servers.

    If you want to use the 11.13 agent version, request the agent media from HP and install the agent on the management server. The agent installation and deployment is described in the HP Operations Agent documentation, which is available from https://softwaresupport.hp.com/group/softwaresupport/search-result?keyword=.

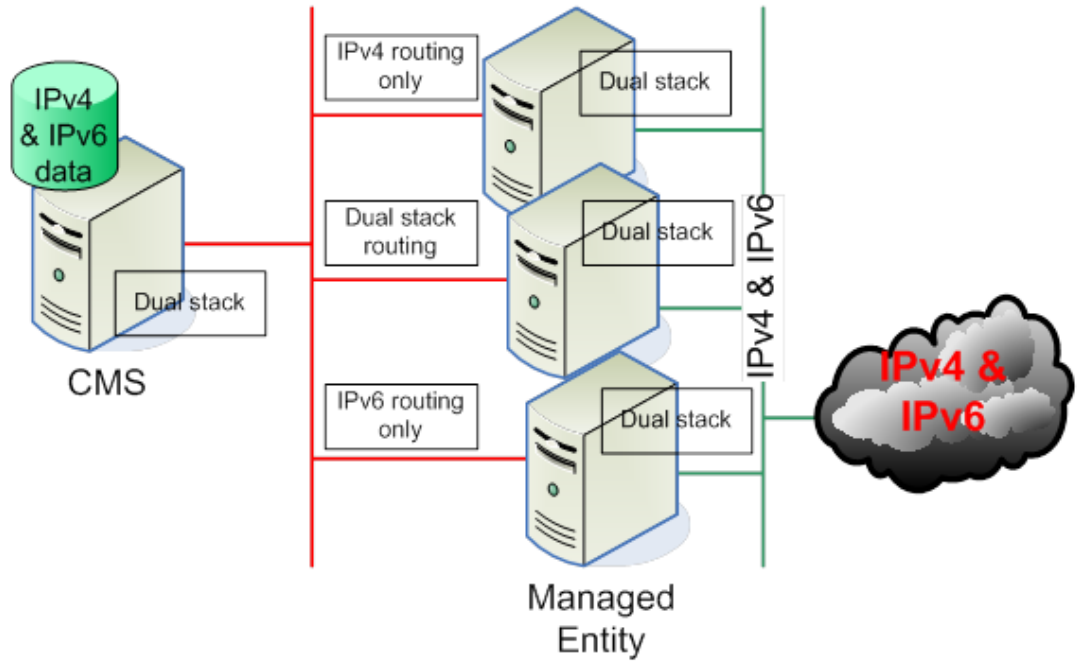
    In addition, download and install the latest HP Operations Agent patches from HP Software Support Online at https://softwaresupport.hp.com.

  - IPv6

  - Dual-stack

    Figure 6 is a graphical presentation of IP network communication with the dual-stack architecture implemented and configured.

Figure 6: IPv4 and IPv6 Coexistence in the Dual-Stack Architecture



HPOM Functionality Supported with IPv6

Enabling the IPv6 protocol in your HPOM environment does not influence the functionality provided with the HPOM software.

Configuring the IPv6 Protocol on the HP Operations Management Server

This section describes the following:

- "IPv6 Configuration Prerequisites" on page 234
- "Enabling IPv6 Support" on page 235
- "Configuring IPv6 in the Cluster Environment" on page 235
- "Checking the IP Version Used in Your HPOM Environment" on page 239

IPv6 Configuration Prerequisites

To use the IPv6 protocol in an HPOM environment, you must first configure IPv6 in your organization on the network and system infrastructure level.

To do so, you must meet the following prerequisites:

- The DNS infrastructure in your organization must support IPv6.
- The network infrastructure in your organization must support and have enabled IPv6 traffic.
- The HP Operations management server must have dual-stack architecture implemented and configured on the operating system level.

  Note: For a graphical presentation of IP network communication in your HPOM environment with the dual-stack architecture implemented and configured, see Figure 6.

- IPv6 traffic must be thoroughly tested before you configure HPOM to use the IPv6 protocol.
- Name resolution for the HP Operations management server should return both IP addresses (IPv4 and IPv6). To check whether this is the case, type the following:

  ```
  opcsvnsl -6 -v -s <management_server_hostname>
  ```

  For example, if you type `opcsvnsl -6 -v -s stenar03.hp.com`, the output should look as follows:

  ```
  Name: stenar03.hp.com
  Addresses: fec0::94f6:cff:fe4d:ccdb, 192.168.1.1
  ```

  Note: You must add both IPv4 and IPv6 addresses to the `/etc/hosts` file on the HP Operations management server.

Enabling IPv6 Support

To enable IPv6 support on the management server, you must perform the following:

- Set the `OPC_IPV6_ACTIVE` configuration variable to `TRUE`:

  ```
  ovconfchg -ovrg server -ns opc -set OPC_IPV6_ACTIVE TRUE
  ```

- Set the `IsIPV6Enabled` Lcore configuration variable to `TRUE`:

  ```
  ovconfchg -ns sec.cm.server -set IsIPV6Enabled TRUE
  ```

- Restart the HP Operations management server and Lcore processes:

  ```
  ovc -start
  ovc -kill
  ```

Configuring IPv6 in the Cluster Environment

This section describes how to configure IPv6 in the cluster environment. The information is organized as follows:

- "Prerequisites" on page 236
- "Installation " on page 236
- "Enabling IPv6 for Server-Agent Communication" on page 236

Prerequisites

Before you configure IPv6 in the cluster environment, make sure the following applies:

- Cluster nodes are configured as IPv4 systems that support the IPv6 network.
- The available IPv6 address for the HP Operations management server virtual node is resolved by DNS.

Installation

To install HPOM to support IPv6 in the cluster environment, follow the procedure:

1. Enable IPv6 on all cluster nodes.
2. Install the HP Operations management server in the same way as on the IPv4-only cluster nodes. All IP addresses during the installation are IPv4. For the installation procedure, see the *HPOM Installation Guide for the Management Server*.
3. When you complete the server installation in the cluster environment, enable IPv6 for communication between server and agent as described in the "Enabling IPv6 for Server-Agent Communication" on page 236.

Enabling IPv6 for Server-Agent Communication

To enable IPv6 for communication between server and agent, perform these steps on the active cluster node:

1. Disable the HP Operations management server monitoring, as follows:

   ```
   /opt/OV/lbin/ovharg -monitor <Server HARG> disable
   ```

2. Follow the appropriate procedure, depending on your cluster environment:

   Veritas cluster

   a. Enable writing in the Veritas cluster configuration:

      ```
      haconf -makerw
      ```

   b. Add a new resource for managing the HP Operations management server IPv6 address:

      ```
      hares -add ov-ipv6 IP ov-server
      ```

   c. Set a network device where the IPv6 address will be applied:

      ```
      hares -modify ov-ipv6 Device <device like eth0>
      ```

   d. Set an IPv6 address:

      ```
      hares -modify ov-ipv6 Address <IPv6 address>
      ```

   e. Set the prefix length for the IPv6 address:

      ```
      hares -modify ov-ipv6 PrefixLen 64
      ```

f. Enable the resource:

```
hares -modify ov-ipv6 Enabled 1
```

g. Check resources for all cluster nodes:

```
hares -probe ov-ipv6 -sys <cluster node>
```

h. Start the resource on the node where the HP Operations management server is active:

```
hares -online ov-ipv6 -sys <cluster node>
```

IPv6 should now be active on the selected cluster node.

i. Set the resource dependency:

```
hares -link ov-application ov-ipv6
```

Sun Cluster

a. Add a new `LogicalHostname` resource to the `HARG` file on the HP Operations management server:

```
clreslogicalhostname create -g <HPOM HARG> -h <IPv6 address> -N <ipmp group>@<cluster node> ... ov-ipv6
```

b. Set the new resource dependency:

i. Get the current resource dependency of the `ov-application` resource:

```
clresource show -y Resource_dependencies ov-application
```

ii. Set a new dependency list by adding the `ov-ipv6` resource:

```
clresource set -y Resource_dependencies=<new dependency list> ov-application
```

Example:

```
clresource set -y Resource_dependencies=ov-ip,ov-ipv6,ov-zpool ov-application
```

Red Hat Cluster (RHCS)

a. Edit `/etc/cluster/cluster.conf` and perform the following changes:

i. Add an IPv6 address inside the resources element:

```
<resources>
    ...
    <ip address="<IPv6 address>" monitor_link="1"/>
</resources>
```

ii. Add an IPv6 address inside the `ov-server` service element:

```
<service autostart="0" domain="ov-server-failover" name="ov-server"
recovery="relocate">
  ...
  <ip ref="<IPv6 address>"/>
```

```
</service>
```

iii. Increase `config_version` by one at the top of the file.

b. Validate the new cluster configuration file by using the `ccs_config_validate` command.

c. Propagate the new cluster configuration to the rest of the cluster nodes:

```
cman_tool version -r Enabling
```

d. Verify `/etc/cluster/cluster.conf` on all cluster nodes.

HP ServiceGuard

a. Reconfigure the cluster to support both IPv6 and IPv4 addresses.

To enable the IPv6 address handling, add an IPv6 subnet to the cluster configuration. To do so, add the following lines to the cluster configuration:

```
SUBNET <IPv6 subnet e.g. fec0:: >
  IP_MONITOR OFF
```

In addition, you might also need to add a heartbeat IPv6 address for each cluster node. Add the following line for each cluster node specification in the cluster configuration:

```
HEARTBEAT_IP <cluster node IPv6 address>
```

Verify and apply the cluster configuration.

b. Reconfigure the server `HARG` file.

Add an HP Operations management server virtual host IPv6 address to the HP Operations management server package configuration by adding the following lines:

```
ip_subnet      <IPv6 subnet e.g. fec0::>
ip_address     <virtual node IPv6 address>
```

Verify and apply the package configuration.

3. After the cluster configuration is completed, enable IPv6 on the server side. To do this, follow these steps:

a. Enable the IPv6 support on the HP Operations management server:

```
ovconfchg -ovrg server -ns opc -set OPC_IPV6_ACTIVE TRUE
```

b. Enable the IPv6 support for SecCM:

```
ovconfchg -ovrg server -ns sec.cm.server -set IsIPV6Enabled TRUE
```

c. Set the IPv4 and IPv6 addresses of the server virtual node as the `SERVER_BIND_ADDR` variable values:

```
ovconfchg -ovrg server -ns bbc.cb -set SERVER_BIND_ADDR <IPv4>,<IPv6>
```

d. Restart the Communication Broker:

```
ovbbccb -stop server <active_local_node>
ovbbccb -start server <active_local_node>
```

e. Check whether the Communication Broker listens for both IPs on the port 383 namespace `bbc.cb`:

   ```
   ovbbccb -status
   ```

f. Enable the HP Operations management server monitoring:

   ```
   /opt/OV/lbin/ovharg -monitor <Server HARG> enable
   ```

Checking the IP Version Used in Your HPOM Environment

You can check the IP version used in your server-agent communication by viewing trace logs and the `System.txt` file on the HP Operations management server. IP addresses used by your HP Operations management server are also visible in the Administration UI or in the output of the `opcnode` command. To obtain this output, type the following:

```
opcnode -list_nodes
```

For instructions on how to use the Administration UI, see the *HPOM Administration UI Online Help*.

IPv6 Protocol Limitations

The following limitations for using the IPv6 protocol apply:

- The HP Operations management server cannot run in an IPv6-only environment. It is required to set a dual-stack architecture.
- HPOM does not support IPv6 “out of the box” when setting up a cluster.

Part II: Processes and Health Monitoring

HP Operations Manager (HPOM) uses certain processes on the management server and on the managed nodes for communication between the server and the nodes, and for communication among processes running on the same server. HPOM also enables you to manage custom processes.

The `opchealth` daemon is a HPOM management server process, which monitors the health of the management server and forwards health status reports to registered clients.

For detailed information about the HPOM processes and health monitoring, see:

- "HPOM Processes" on page 241
- "HPOM Health Monitoring" on page 260

Chapter 6: HPOM Processes

In this Chapter

This chapter provides a functional overview of the processes used by HP Operations Manager (HPOM) on the management-server processes and managed-node. For example, you can learn about the processes used by the message, monitor, and action agents on the managed node, and understand how they communicate with the message and action managers on the management server. You can also find out about how the management server communicates with the remote hosts running Java clients.

The information in this chapter covers the following high-level topics:

Communication Flows in HPOM

HP Operations agents and management servers communicate through Remote Procedure Calls (RPCs) based on BBC, queues, pipes, or signals. The mechanisms apply to communication between the management server and the managed nodes, as well as to communication between processes running locally on the management server.

Figure 7: Functional Overview of HPOM

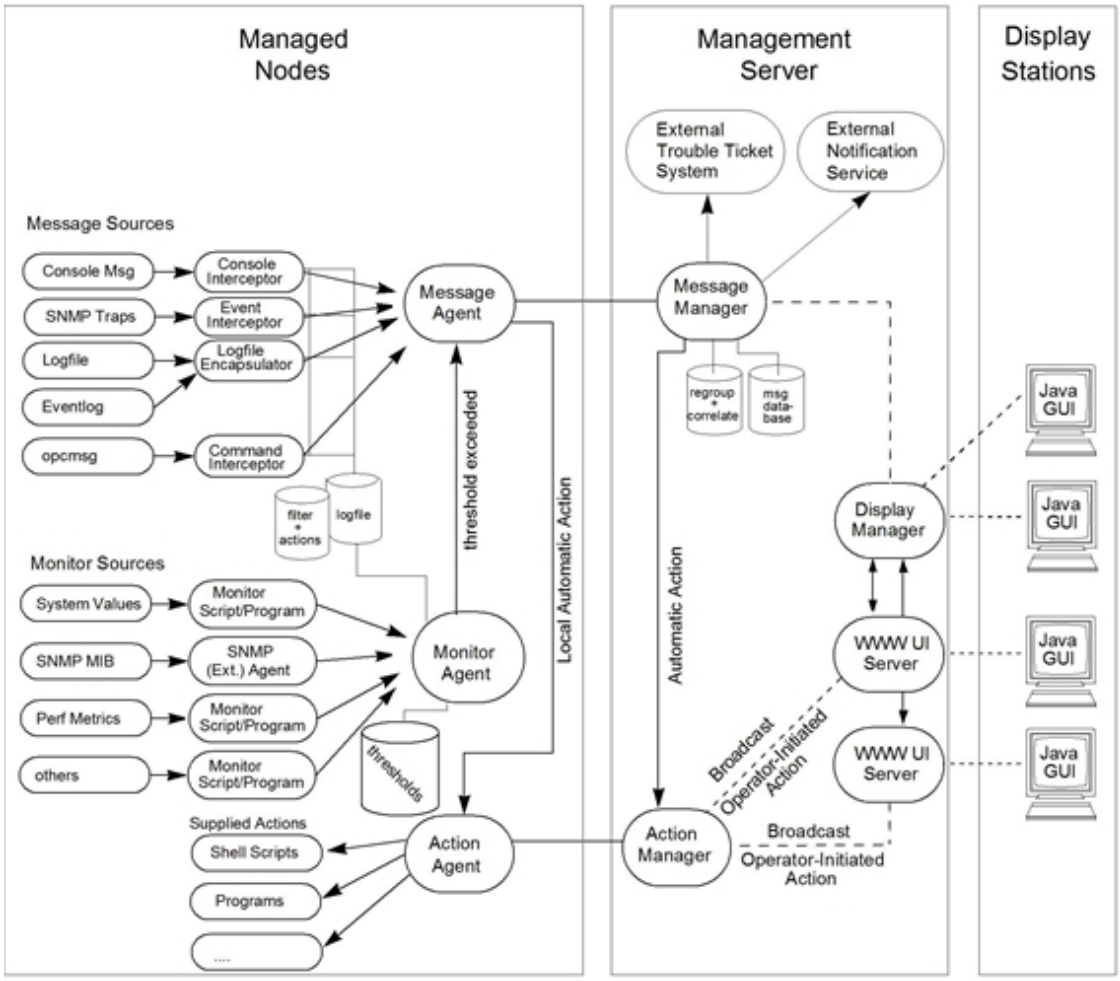


Figure 7 illustrates the communication flow between the HPOM-specific processes running on the management server and the managed nodes.

For more information on how the processes communicate with one another and what each process does, see "HPOM Management Server Processes" on page 242 and "HPOM Managed Node Processes" on page 246.

HPOM Management Server Processes

This section describes the HPOM processes and their associated files on the management server. In this section, you can find detailed information about the following topics:

- "Processes on the HPOM Management Server" on page 243
- "Process Files on the HPOM Management Server" on page 245

Processes on the HPOM Management Server

This following list describes the processes that run on the HP Operations management server. For more information about the queue files and pipes that the listed processes use, see "Process Files on the HPOM Management Server" on page 245.

`opcactm`	Action manager that feeds the action agents with automatic actions, operator-initiated actions, scheduled actions, and application startup and broadcasting information through the control agent. In addition, external instructions are determined using this mechanism.
`ovoareqsdr`	Request sender that informs the control agents to start, stop, or update their local HPOM agents. The request sender is also responsible for the self-monitoring of HPOM manager services, and for the heartbeat-polling of the managed nodes.
`ovcd`	Control daemon that controls and checks the status of processes and components, which are registered with it.
`opcdispm`	Display manager that serves Java GUIs. The display manager also feeds the action manager with operator-initiated actions, application startup information (not requiring a separate terminal), and broadcasting information issued by operators. It also serves clients connected to the MSI for message and configuration changes. Several Java GUI clients may be active at the same time.
`opcbbcdist`	Configuration management adapter between the HP Operations management server and the HTTPS agents that creates instrumentation from existing actions, commands, and monitors, and switches `nodeinfo` settings into the XPL format used on HTTPS nodes.
`opcecm`	Event correlation manager that connects to the server MSI to allow access to and modification of messages from the HPOM message flow by the event correlation (EC) engine. Depending on filters and conditions, the messages are then correlated and written back to HPOM. The messages display in the `Message Details` window (available from the Message Browser) with the message source MSI `opcecm`. Like all server processes, the event correlation manager is controlled by the OV Control, `ovcd`.
`opcecmas`	Annotation server that runs on the management server and obtains data from outside the ECS engine for use within correlation circuits. This process connects to the `opcecm` process using the standard annotate API. It receives

	annotate requests for launching external programs and returns the output to the circuit.
opcmsgm	Message manager that receives messages from the managed nodes through the message receiver (opcmsgrb). The messages can be correlated, regrouped and logged by the message manager running on the management server. The message manager is also responsible for adding annotations, triggering notifications, and forwarding the message to the trouble ticket and notification service manager for external notification and trouble ticket generation.
opcforwm	Message forwarding manager that relieves the message manager, opcmsgm, of time-consuming tasks (for example, sending messages to remote managers). This relief allows the message manager to manage messages more effectively. On the local “source” management server, the message forwarding manager receives data from the message manager (in the form of messages), the action manager (action responses), and the display manager (message operations such as acknowledge, add annotation, and so on). The message forwarding manager sends data to the message receiver on the “target” management servers.
opctss	Distribution manager subprocesses that transfer configuration data to the distribution agent through TCP/IP.
opcttnsm	Trouble-ticket and notification-service manager that feeds the external notification and trouble-ticket interfaces with message attributes. This manager is an auxiliary process of the message manager designed to ensure high message throughput. If external instructions are specified for a message, the trouble ticket and notification service manager evaluates the help text through the action manager. Whenever the trouble ticket and notification service manager receives a message in its queue, it passes the message on to the trouble ticket interface or the external notification service. It does so by forking and executing the customer-defined program that receives the message (that is, the ticketing interface or the notification service). As soon as this program is finished and exited, a SIGCHLD is sent to the trouble ticket and notification service manager. The manager stops processing the message queue until it receives another SIGCHLD.
opcuiwww	Server process that serves the HPOM Java-based operator GUI. This process forwards all communication requests between the Java GUI and the display

	manager. For each Java GUI, at least one server process is started.
opcuihttps	Server process that acts as a proxy between the Java GUI client and the HPOM management server using the HTTPS protocol.
opcsvcm	Service engine that maintains the global (operator-independent) service status and can log service changes into the database. By default, remote access to the service engine is disabled.
opcsvcdisc	Service discovery process that receives discovered or forwarded topology data from discovery agents or other management servers. The discovery server uses context mapping rules to filter the topology data before storing it in the database.

Process Files on the HPOM Management Server

The files used by the HPOM management-server processes reside in the following directory:

`/var/opt/OV/share/tmp/OpC/mgmt_sv`

The following list describes the queue files and pipes that the HPOM management-server processes use. For more information about the HPOM management-server processes themselves, see "Processes on the HPOM Management Server" on page 243.

actreqp/actreqq	Queue or pipe used by the display manager, message manager, TTNS manager, (and action manager) to pass action requests to the action manager.
actrespp/actrespq	Queue or pipe used by the message receiver, request sender, and action manager to pass action responses to the action manager.
ctrlq/ctrlp	Queue or pipe between the display manager and control manager.
forwmgrp/forwmgrq	Queue or pipe used by the message manager, display manager, action manager, and the forward manager to pass data to be forwarded to other management servers.
magmgrp/magmgrq	Queue or pipe between the message dispatcher and the request handler.
mpicdmp/mpicdmq	Queue or pipe used by the display manager and the message stream interfaces to transfer control sequences for message-change event handling.
mpicmmp/mpicmmq	Queue or pipe used by the message manager and message-stream interfaces to transfer control sequences for message handling through the Message-Stream Interface (MSI).

mpimmp/mpimmq	Queue or pipe used by the message manager and the message-stream interfaces to transfer messages from MSI programs to the message manager.
msgmgrq/msgmgrp	Queue or pipe between the message receiver and message manager.
opcecap/opcecaq	Queue or pipe used to pass messages from the message manager to the event correlation manager.
pids	Process IDs of the HPOM managers that are controlled by the HPOM control manager, which is also used for self-monitoring.
rqsdbf	Buffer file used by the request sender to store requests if the control agent on a given managed node cannot be accessed
rqsp/rqsq	Queue or pipe between the request handler and the request sender. Also used by the display manager and the action manager
ttnsarp/ttnsarq	Queue or pipe used by the trouble-ticket manager and action manager when message instructions have to be fetched by the trouble-ticket notification-service (TTNS) manager.
ttnsq/ttnsp	Queue or pipe between the message manager, trouble-ticket manager, and notification-service manager.
WAP*	Queue or pipe used by opcuiwww for application responses.
WWW*	Queue or pipe used by opcuiwww for message change events.

HPOM Managed Node Processes

This section describes the HPOM processes and their associated files on the managed nodes. In this section, you can find detailed information about the following topics:

Processes on the Managed Node

The information in this section lists and describes the HPOM processes on the managed node. For more information about the files the managed node processes use, see "Process Files on the Managed Node" on page 248.

`coda`	Embedded performance component that collects performance counter and instance data from the operating system. Threshold monitor policies are used to access performance metrics collected by the embedded performance component.
`opcacta`	Action agent that is responsible for starting and controlling automatic actions, operator-initiated actions, and scheduled actions (that is, scripts and programs). The action agent is also used for broadcasting commands and for launching applications configured as Window (Input/Output).
`opceca`	Event-correlation agent that connects to the agent MSI in the same way that the ECS runtime library is integrated into the HPOM server. This connection allows access to (and modification of) messages from the HPOM message flow on the agent. The messages modified by this process display in the `Message Details` window (available from the Message Browser) with the message source "`MSI: opceca`". Like all agent processes, `opceca` is controlled by the control agent.
`opcecaas`	Annotation server that runs on a managed node and obtains data from outside the ECS engine for use within correlation circuits. This process connects to the `opceca` using the standard annotate API. It receives annotate requests for launching external programs and returns the output to the circuit.
`opcle`	Log-file encapsulator that scans one or more application or system log files (including the Windows Eventlog) for messages or patterns specified by the HPOM administrator. The log-file encapsulator forwards the scanned and filtered messages to the message agent.
`opcmona`	Monitor agent that monitors the following components: • System parameters, for example: CPU load, disk utilization, and kernel parameters. • SNMP MIBs • Other parameters, if specified The monitor agent checks the values it finds against predefined thresholds. If a threshold is exceeded, a message is generated and forwarded to the message

	agent. The polling interval of the monitored object can be configured by the HPOM administrator. In addition, the `opcmon(1)` command and `opcmon(3)` API can be used (asynchronously) to feed the monitor agent with the current threshold values. The monitor agent does not immediately begin monitoring when agents are started. Instead, it waits one polling interval, and only then executes the monitor script for the first time. Typically, polling intervals are 30 seconds to 5 minutes.
`opcmsga`	Message agent that receives messages from the log-file encapsulator, monitor agent, console interceptor, event interceptor and message interceptor on the local system. The messages are forwarded to the message receiver running on the management server. If the connection to the management server has been lost, the messages are buffered locally. The message agent triggers local automatic actions by forwarding the task to the action agent.
`opcmsgi`	Message interceptor that receives and processes incoming messages. The `opcmsg(1)` command and `opcmsg(3)` API can be used to forward messages to HPOM. Conditions can be set up to integrate or suppress chosen message types.
`opcctla`	Control agent that starts and stops all HPOM agents, and performs HPOM self-monitoring tasks. The control agent is informed of new configuration and distribution requests by the request sender.
`opctrapi`	Event interceptor that is the message interface for feeding SNMP events to HPOM. Conditions can be set to integrate or suppress selected message types.

Process Files on the Managed Node

This section describes the pipes and queue files used by the HPOM processes outlined in "Processes on the Managed Node" on page 247. The locations of these process files are listed in "Location of Process Files on the Managed Node" on page 249.

`actagtp/actagtq`	Queue or pipe for pending action requests for the action agent. The pending action requests are filled by the message agent and the control agent. The action agent polls the queue every 5 seconds.
`monagtq/monagtp`	Queue on UNIX systems between the HPOM monitor command `opcmon(1)`, the HPOM monitor API `opcmon(3)`, and the monitor agent. The monitor agent checks the queue after the termination of the triggered monitor scripts or

	programs every 15 seconds, if externally monitored objects are configured.
`mpicmap/mpicmaq`	Queue or pipe used by the message agent and the message stream interfaces to transfer control sequences for message handling through the MSI.
`mpimap/mpimaq`	Queue or pipe used by the message agent and the message stream interfaces to transfer messages from MSI programs to the message agent.
`msgagtdf`	File that holds any messages that cannot be passed to the management server (for example, if the network is down). The messages are read from this file after the management server is available.
`msgagtp/msgagtq`	Queue or pipe for local buffering of messages to be sent to the message receiver when the management server is not accessible.
`msgip/msgiq`	Queue or pipe (only on UNIX systems) between the HPOM message command `opcmsg(1)` or the HPOM message API `opcmsg(3)` and the message interceptor.
`opcecap/opcecaq`	Queue or pipe that passes messages from the message agent to the event-correlation agent.
`pids`	Process IDs of HPOM agents controlled by the control agent.
`trace` (plain text)	HPOM trace log file.
`aa*`	Temporary files used by the action agent, for example, to store the action or application output written to `stderr` and `sdtout`.
`moa*`	Temporary files used by the monitor agent.

Location of Process Files on the Managed Node

Table 11 shows the location of the files used by the HPOM agent processes described in "Processes on the Managed Node" on page 247. For more information about the files used by the HPOM agent processes on the managed nodes, see "Process Files on the Managed Node" on page 248.

Table 11: Locating Process-related Files on the Managed Nodes

Platform	File Location
AIX	`/var/lpp/OV/tmp/OpC`

Locating Process-related Files on the Managed Nodes, continued

Platform	File Location
HP-UX 11.x	/var/opt/OV/tmp/OpC
Linux	
Solaris	
Windows	\usr\OV\tmp\OpC*<node>*

HPOM-Agent Configuration Files

Table 12 describes the files you can use to configure the HPOM agent, and indicates whether the contents of the files are encrypted. For more information about the locations of the agent-configuration files, see Table 13.

Table 12: Agent Configuration Files and their Contents

File	Contents	Encrypted?
le	Log-file encapsulation configuration	✓
mgrconf	Flexible-management configuration file	✗
monitor	Monitor agent policy file	✓
msgi	Message interceptors opcmsg(1) and opcmsg(3)	✓
primmgr	Flexible-management configuration file.	✗
trapi	SNMP event interceptor.	✓

Location of HPOM Agent Configuration Files

Table 13 lists the locations of the HPOM agent specific configuration files described in Table 12.

Table 13: Locating Agent Configuration Files on the Managed Nodes

Platform	Agent File Location
AIX	/var/lpp/OV/conf/OpC

Locating Agent Configuration Files on the Managed Nodes, continued

Platform	Agent File Location
HP-UX 11.x	/var/opt/OV/conf/OpC
Linux	
Solaris	
Windows	\usr\OV\conf\OpC*<node>*

Process Registration

The HPOM process control component (ovcd) controls all HPOM management server processes and ensures they are started and stopped in the order that is defined by the Dependency element[1]. For details, see the description of Dependency on page 253.

Each server process is registered with the process control component with one XML registration file. The default registration files can be found at the following locations:

- For server components:

  /etc/opt/OV/share/ovc
- For agent components:

  /opt/OV/misc/eaagt

All HPOM processes are automatically registered with the HPOM control daemon, ovcd, and can be controlled by using the ovc command with the -start, -stop, and -status options respectively. Each HPOM server process has its own XML registration file that defines how the HPOM processes are handled.

The configuration files that define the registration process for each process are stored at the following location on the management server and the managed node:

<*OvDataDir*>/conf/ctrl

On the management server, the ctrl directory contains registration files for all HPOM processes—both management server processes and managed node processes if the management server is also configured as a managed node. On the managed node, the directory contains registration files only for the managed node processes.

[1]The order in which MSI applications receive messages is not defined by this order.

Custom Process Management

HPOM enables you to manage custom processes by adding them to a list of managed components and registering them with the HPOM control daemon, `ovcd`. In this way, additional custom processes can be managed in the same way as any other HPOM process.

To add a custom component to HPOM control, create an XML registration file for this component. You can use the `opccustproc1.xml` sample file that is provided with HPOM as a template for your XML registration file.

Sample XML Registration File

The `opccustproc1.xml` sample file that is provided with HPOM as a template for your XML registration file can be found at the following location:

`/opt/OV/contrib/OpC/opccustproc`

The syntax of this file is the following (with the default values):

```
<?xml version='1.0' encoding='UTF-8' standalone='yes'?>
<ovc:OvCtrl xmlns:ovc="http://openview.hp.com/xmlns/ctrl/registration/1.5">
  <ovc:Component>
    <ovc:Name>ComponentName</ovc:Name>
    <ovc:Label>
      <ovc:String>ComponentLabel</ovc:String>
    </ovc:Label>
    <ovc:Category>Category</ovc:Category>
    <ovc:Options>
      <ovc:AllowAttach>false</ovc:AllowAttach>
      <ovc:AutoRestart>false</ovc:AutoRestart>
      <ovc:AutoRestartLimit>5</ovc:AutoRestartLimit>
      <ovc:AutoRestartMinRuntime>60</ovc:AutoRestartMinRuntime>
      <ovc:AutoRestartDelay>5</ovc:AutoRestartDelay>
      <ovc:MentionInStatus>true</ovc:MentionInStatus>
      <ovc:Monitored>true</ovc:Monitored>
      <ovc:StartAtBootTime>true</ovc:StartAtBootTime>
      <ovc:CoreProcess>false</ovc:CoreProcess>
      <ovc:IsContainer>false</ovc:IsContainer>
      <ovc:AutoShutdown>false</ovc:AutoShutdown>
      <ovc:AutoShutdownTimer>1</ovc:AutoShutdownTimer>
      <ovc:PollingInterval>30</ovc:PollingInterval>
    </ovc:Options>
    <ovc:ProcessDescription>ProcessDescription</ovc:ProcessDescription>
    <ovc:CommandLine>CommandLine</ovc:CommandLine>
    <ovc:OnHook>
      <ovc:Name>OnHookName</ovc:Name>
      <ovc:Actions>Actions</ovc:Actions>
```

```
    </ovc:OnHook>
    <ovc:OnEvent>
      <ovc:Name>OnEventName</ovc:Name>
      <ovc:EventOptions>EventOptions</ovc:EventOptions>
      <ovc:Actions>Actions</ovc:Actions>
    </ovc:OnEvent>
</ovc:Component>
</ovc:OvCtrl>
```

In the context of HPOM control, a component is an entity that can be started, stopped, or notified (performing an action in response to an event). The component consists of the following elements:

`Name` *(required)*:	Each component has a unique name that is used to address it. The name is an ASCII identifier and it is not localized.
`Label` *(required)*:	Each component has a label that is used when printing the status of a component. For example, the `opcle` component would have a label “Logfile Encapsulator”. The label can be localized.
`Description` *(optional)*:	Each component can have a text description. The description can be localized.
`Dependency` *(optional)*:	Each component can have dependencies to other components. Dependencies are used when the `start` or `stop` command is issued. For example, when the logfile encapsulator has a dependency on the `opcapm` component, this component must be started before the logfile encapsulator. If a component is stopped, which is defined elsewhere as a dependency, the affected dependant is stopped first.
`Category` *(optional)*:	Each component can belong to none, one, or more categories. A category is a way to group components together to make operations easier (interfaces of HPOM control allow operations on components directly or on category grouping). The category is not localized.
`Options` *(optional)*:	Each component can have the following options: • `AllowAttach` (`TRUE/FALSE`): Instructs HPOM control not to kill the component if it is already started, but just to attach to it (meaning that the component does not have to be stopped first). The default is `FALSE`. • `AutoRestart` (`TRUE/FALSE`): Restarts the component if it terminates unexpectedly. The default is `FALSE`. If you set this option to `TRUE`, you enable the following options: ◦ `AutoRestartLimit`: Specifies the maximum number of component automatic restarts. The default is 5.

	• `AutoRestartMinRuntime`: Specifies how long in seconds the component must run before it can be restarted automatically. The default is 60. • `AutoRestartDelay`: Specifies after how long in seconds the component is restarted automatically. The default is 5 seconds. • `MentionInStatus` (`TRUE/FALSE`): Specifies whether the status of the component is included in the status report. The default is `TRUE`. • `Monitored` (`TRUE/FALSE`): Specifies whether the errors are reported when a component terminates unexpectedly. The default is `TRUE`. • `StartAtBootTime` (`TRUE/FALSE`): Specifies whether the component should be started at a boot time. Evaluated only when `-boot` is specified. The default is `TRUE`. • `CoreProcess` (`TRUE/FALSE`): Specifies that the component should be stopped only if the `-kill` option is used. The default is `FALSE`. • `IsContainer` (`TRUE/FALSE`)[1]: Specifies whether the component should be treated as a container for other components. The default is `FALSE`. • `AutoShutdown` (`TRUE/FALSE`): Specifies whether the container should be stopped when all its contained components are stopped. The default is `FALSE`. • `AutoShutdownTimer`: Specifies after how long in seconds the container component is stopped when all of its contained components are stopped. The default is 30. • `Container`: Defines a name of the container for the contained component. • `PollingInterval`: Defines how often in seconds the container is polled to obtain the run status of the contained component. The default is 30. • `WorkingDirectory`: Specifies the working directory for the component (that is, the directory in which the component operates by default). Keep in mind, however, that depending on the internal operation of the component, the location of the actual working directory may differ from the specified one.
`ProcessDescription` *(required)*:	Each component has a process description that is used as follows: the process name, which is part of the operating system process table, is used

[1]For HP internal use only. Leave the default value.

	to connect to the component in `OvCtrl` by comparing the string from the process table (that is, the process name) with the string value from the `ProcessDescription` element. *HP-UX PA-RISC only:* Because of system limitation, the length of the process string is limited to 14 characters.
`CommandLine` *(optional)*:	Allows matching components against their command lines, not just against the process description. It makes it possible to distinguish between different Java virtual machines running on a system. Matching is done by using regular expressions.
`OnHook` (optional):	A component is usually a process and has a lifetime. Within its lifetime, the component can be in the following states that are known to HPOM control: stopped, starting, initializing, running, or stopping. Several hooks allow the component to register actions that are executed and affect the state changes of a process. A hook is defined by `Name` (required and predefined) and `ActionType` (required). The following hooks are available: **Note:** For very simple components, define only the `START` action. • `START_CHECK`: Allows defining a sequence of actions that must complete successfully before starting the component. Can be used to conditionally start the component. • `START`: Specifies the start sequence of the component. • `INITIALIZE`: Allows specifying additional actions that must complete successfully before the component is considered running. • `STOP`: Specifies the way in which the component is stopped. If it is not specified, HPOM control tries to stop the component in the way that is the default for the operating system. **Caution:** It is not recommended to use the Execute action in a `STOP` hook on Windows. On Windows, during the shutdown process, it is not allowed to start a new process and as a result the action will fail. • `CHECK_STATUS`: Specifies the status check sequence of a contained component.
`OnEvent` *(optional)*:	Specifies what must be done when an event is received. The event is defined by the following elements:

	• `Name` *(required)*: A name of the event to register for. It can be also an arbitrary string (the ":" is used to specialize the string into event:subevent). • `EventOptions` *(optional)*: Triggers additional processing when the event is received. Currently, the following two options are defined: • `ReevaluateStart`: Starts the component if it is not running in the `START_CHECK` sequence of actions. This is necessary if the event potentially impacts the startup condition of the component. • `ReevaluateStop`: Stops the component if it is running and the `START_CHECK` sequence of actions fails. This is necessary if the event potentially impacts the startup condition of the component. • `ActionType` *(required)*: Specifies one or more actions to be executed when an event is received. Different types of actions are supported (not all actions are relevant for each event). The following predefined events are available: • `CHECK_POLICY`: This event is sent when a policy is changed (added or modified). • `REMOVE_POLICY`: This event is sent when a policy is removed. • `FIRST_POLICY: <policy_type>`: The first policy of a given type is installed for the first time. • `LAST_POLICY: <policy_type>`: The last policy of a given type is removed. • `ENABLE_POLICY`: A policy is enabled. • `DISABLE_POLICY`: A policy is disabled. `<ovc:OnEvent>` ` <ovc:Name>DISABLE_POLICY:mgrconf</ovc:Name>` ` <ovc:EventOptions>` `  <ovc:ReevaluateStart>false</ovc:ReevaluateStart>` `  <ovc:ReevaluateStop>false</ovc:ReevaluateStop>` ` </ovc:EventOptions>`
`ActionType` *(required)*:	You can specify more than one `ActionType` for `OnHook` and `OnEvent`. The actions are processed one after another and all must complete successfully for the event or hook to be considered successful. The following actions are available:

	• `Execute`: Runs a command and waits for it to complete. This action is meant for processes that daemonize themselves. You can also specify environment. On Windows, the `EXE` extension for the command is not required. • `Start`: Runs a command similarly to `Execute` except that it returns as soon as the process is spawned. Use it in actions to start processes that do not daemonize themselves. With the `Start` action, it is possible to put a process in the background. On Windows, the `EXE` extension for the command is not required. • `UXSignal` *(Unix only)*: Sends a signal to the component. The signal name can be specified. Keep in mind that different operating systems use different signals. For details, see the relevant operating system documentation. • `WinEvent` *(Windows only)*: Sends events on Windows. Windows does not have the signal mechanism but uses events for IPC.

Examples of XML Registration File Configuration

The following are examples of XML registration file configuration:

OnHook Component

```
<ovc:OnHook>
   <ovc:Name>START</ovc:Name>
      <ovc:Actions>
         <ovc:Start>
            <ovc:CommandLine>START-CommandLine</ovc:CommandLine>
         </ovc:Start>
      </ovc:Actions>
</ovc:OnHook>
```

OnEvent Component

```
<ovc:OnEvent>
   <ovc:Name>RECONFIGURE</ovc:Name>
   <ovc:EventOptions>
      <ovc:ReevaluateStart>false</ovc:ReevaluateStart>
      <ovc:ReevaluateStop>false</ovc:ReevaluateStop>
   </ovc:EventOptions>
   <ovc:Actions>
      <ovc:UXSignal>
         <ovc:Name>SIGUSR1</ovc:Name>
      </ovc:UXSignal>
```

```
   </ovc:Actions>
</ovc:OnEvent>
```

Adding a Custom Component to HPOM Control

To register a custom process with the HPOM control daemon, `ovcd`, follow these steps:

1. Create an XML registration file to register the custom process.

   You can use the `opccustproc1.xml` sample file that is provided with HPOM as a template for your XML registration file, as follows:

   a. Copy and rename the template configuration file `/opt/OV/contrib/OpC/opccustproc/opccustproc1.xml` according to your needs. For example:

   ```
   # cp /opt/OV/contrib/OpC/opccustproc/opccustproc1.xml
   /opt/OV/contrib/OpC/opccustproc/<my_process>.xml
   ```

   Note that you must replace *<my_process>* with the name of the process you want to register.

   b. Modify the following tags in the *<my_process>*`.xml` file according to your needs. For further help, see the example for the `opccustproc1.xml` file:

   ```
   <ovc:Name>opccustproc1</ovc:Name>
   <ovc:Label>
   <ovc:String>OMU Custproc 1</ovc:String>
   </ovc:Label>
   <ovc:AllowAttach>false</ovc:AllowAttach>
   <ovc:AutoRestart>true</ovc:AutoRestart>
   <ovc:AutoRestartLimit>5</ovc:AutoRestartLimit>
   <ovc:AutoRestartMinRuntime>60</ovc:AutoRestartMinRuntime>
   <ovc:AutoRestartDelay>5</ovc:AutoRestartDelay>
   <ovc:MentionInStatus>true</ovc:MentionInStatus>
   <ovc:Monitored>true</ovc:Monitored>
   <ovc:StartAtBootTime>false</ovc:StartAtBootTime>
   <ovc:WorkingDirectory>/var/opt/OV/share/tmp/OpC/mgmt_
   sv</ovc:WorkingDirectory>
   <ovc:ProcessDescription>opccustproc1</ovc:ProcessDescription>
   ```

   Under the `<ovc:Name>START</ovc:Name>` tag of the `OnHook` element, replace the `CommandLine` tag with the program that you want to start. For example:

   ```
   <ovc:OnHook>
      <ovc:Name>START</ovc:Name>
         <ovc:Actions>
            <ovc:Start>
               <ovc:CommandLine>/opt/OV/bin/OpC/opccustproc1</ovc:CommandLine>
            </ovc:Start>
         </ovc:Actions>
   </ovc:OnHook>
   ```

You can delete the <ovc:Name>**START_CHECK**</ovc:Name> tag of the OnHook element, along with its subtags:

```
<ovc:OnHook>
   <ovc:Name>START_CHECK</ovc:Name>
      <ovc:Actions>
         <ovc:Execute>
           <ovc:CommandLine>/opt/OV/bin/OpC/opcsv -
startable</ovc:CommandLine>
         </ovc:Execute>
         <ovc:Execute>
            <ovc:CommandLine>/opt/OV/bin/OpC/opcsv -available
opccustproc1</ovc:CommandLine>
         </ovc:Execute>
      </ovc:Actions>
</ovc:OnHook>
```

2. Check the syntax of the new *<my_process>*.xml file by using the ovcreg(1) command with the -check parameter. Run the following command:

   ```
   # ovcreg -check /opt/OV/contrib/OpC/opccustproc/<my_process>.xml
   ```

   The location of the ovcreg registration tool is the following:

   ```
   <OvInstallDir>/bin
   ```

3. Register the new *<my_process>*.xml file by using the ovcreg command with the -add parameter. Run the following command:

   ```
   # ovcreg -add /opt/OV/contrib/OpC/opccustproc/<my_process>.xml
   ```

4. Start, stop, and check the status of the new custom process by using the ovc command with the -start, -stop, and -status parameters respectively.

   For example, to start the custom process, run the following command:

   ```
   # ovc -start <my_process>
   ```

5. To cancel the registration of the custom process, use the ovcreg command with the -del(ete) parameter.

   For example, run the following command:

   ```
   # ovcreg -del opccustproc1
   ```

Chapter 7: HPOM Health Monitoring

In This Chapter

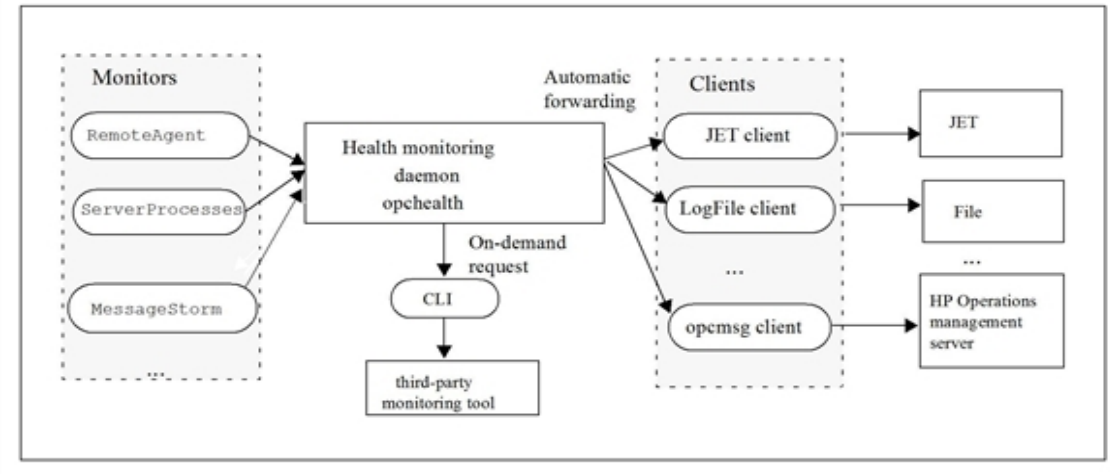
This chapter describes which features you can use for monitoring the health of the HP Operations management server. These features are as follows:

Health Monitoring

Health monitoring consists of health monitoring and health status forwarding. The health of the HP Operations management server is monitored by registered monitors (that is, special tools for monitoring the status of a particular resource) and the health status is forwarded to all registered clients.

Health monitoring is performed by the `opchealth` daemon that runs as part of the HP Operations management server processes, which are controlled by the OV Control daemon. In general, the health monitoring daemon collects health data from registered monitors, filters it, and then forwards it automatically to registered clients. The registered client sends health data either to a file, a database, or a remote application (depending on the type of registered client).

Figure 8: Health Monitoring Flow



Health Monitoring Basics

When monitoring the health of the HP Operations management server and forwarding the health status, you must be familiar with all the options that are available with the `opchealth` tool, the location of the health configuration directories, how health data is divided, how to enable or disable tracing, and the location of the log file.

Health Monitoring Tool

To monitor the health of the HP Operations management server and forward the health status, use the `opchealth` tool that you can find at the following location:

`/opt/OV/bin/OpC/`

The syntax of the `opchealth` tool is as follows:

```
opchealth -disable|-enable
          -health
          -status
          -issues
          -syntax
          -report [-health|-status|issues]
          -register client|monitor <configuration_file>
          -unregister client|monitor <name>
          -clients
          -monitors
          -trace [enable|disable]
          -h|-?|-help
```

You can use the following options with the `opchealth` tool:

`-disable|-enable`	Enables or disables health monitoring and forwarding the health status.
`-health`	Returns general health information.
`-status`	Returns status information from all monitored sources.
`-issues`	Returns a list of all current issues.
`-syntax`	Returns the syntax of statuses and issues as well as all possible issues and statuses provided by registered monitors.
`-report`	Returns the required health data in a readable format.
`-register client|monitor <monitor_conf_file>`	Registers a client or a monitor.

`-unregister client|monitor <monitor_name>`	Unregisters a client or a monitor.
`-clients`	Returns a list all registered clients.
`-monitors`	Returns a list of all registered monitors.
`-trace`	Enables or disables health tracing.
`-h|-?|-help`	Shows the usage.

Note: If no option is used with the `opchealth` tool, the health monitoring and forwarding status is shown (that is, if health monitoring is enabled or disabled).

For detailed information about all `opchealth` options, see the *opchealth* manual page.

Location of the Health Configuration Directories

You can find the basic health configuration directory, `HEALTH_CONF_DIR`, at the following location:

`/etc/opt/OV/share/conf/OpC/mgmt_sv/health/`

The `monitors` subdirectory contains configuration files for registered monitors and the `clients` subdirectory contains configuration files for registered clients.

The temporary location of the health configuration directory, `HEALTH_TEMP_DIR`, is as follows:

`/var/opt/OV/share/tmp/OpC/health/`

Health Data

Health data consists of the following:

- General health information

  Represents the general health of the HP Operations management server. The general health rates range from 1 through 10 and are grouped as follows: `CRITICAL` (`1`–`3`), `MAJOR` (`4`–`6`), `MINOR` (`7`–`9`), and `OK` (`10`).

  The record of the general health looks as follows:

  <general_rate_group>:*<general_rate>*:
  <previous_general_rate_group>:<>*previous_general_rate*>:

  For example:

  `CRITICAL:1:OK:10:`

  The general health status is shown as the worst status of any monitored source (disregarding the status of all other monitored sources).

- Status of a specific monitored source

Represents the status of each monitored source.

The record is shown as a list containing the statuses of all monitored sources. For each monitored source, the record looks as follows:

<monitored_source_name>:*<status_value>*:*<rate>*:
<monitored_source_specific_data>:[1]

For example:

```
SERVER_TRACING:NO:10:
PROCESS_DOWN:YES:1:opcmsgm:
NODE_DOWN:YES:2:5:
```

The typical status value is `YES` or `NO`, but other values are also possible (for example, `OK` or `FAILED`).

The following is an example of the syntax:

```
AGENT_BUFFERING:YES|NO:<rate>:<number_of_agents>:
AGENT_FAILURE:YES|NO:<rate>:<number_of_agents>:
AGENT_DOWN:YES|NO:<rate>:<number_of_agents>:
NODE_DOWN:YES|NO:<rate>:<number_of_agents>:
```

- Current issues

  Represents a problem on the HP Operations management server.

  The record is shown as a list containing all current issues. For each issue, the record looks as follows:

  <monitored_source_name>:*<rate>*:*<issue_data>*:

  For example:

  ```
  PROCESS_DOWN:1:opcdispm:
  ```

  The following is an example of the syntax:

  ```
  AGENT_BUFFERING:<rate>:<node>:
  AGENT_FAILURE:<rate>:<node>:
  AGENT_DOWN:<rate>:<node>:
  NODE_DOWN:<>rate>:<node>:
  ```

Each status or issue has a rate that specifies how critical a certain status or issue is for the health of the HP Operations management server. The rates are grouped as follows: `CRITICAL` (`1`–`3`), `MAJOR` (`4`–`6`), `MINOR` (`7`–`9`), and `OK` (`10`). Low rates (that is, close to `1`) mean that a certain HP Operations management server component does not function and may block the management server. Middle rates (that is, close to `5`) mean that a certain HP Operations management server component does not function properly, but this does not have a critical influence on the performance of the HP Operations management server. High rates (that is, close to `10`) mean that there may be a minor issue with an HP Operations management server component.

[1] *<monitored_source_specific_data>* is optional because it depends on the monitor.

Tracing and Logging

The health monitoring daemon does not produce any tracing data by default. Therefore, to activate or deactivate tracing, use the `-trace [enable|disable]` option.

Tracing data is stored in the following file:

`$HEALTH_TEMP_DIR/trace.log`

In addition, health event logging provides a standard way of recording important health events. When troubleshooting, you can analyze the following log file to find out when and where new events occurred:

`/var/opt/OV/share/tmp/OpC/health/health.log`

Health Monitoring

The basic function of health monitoring is to collect health data from the registered monitors that monitor a specific functionality of the HP Operations management server (for example, management server processes, the remote agent status, the system status, and so on). The monitor in this case acts as a plugin for the health monitoring daemon.

The health monitoring daemon contains a configuration file with an internal list of the registered monitors, `$HEALTH_CONF_DIR/HMI_monitors.conf`, in which each line represents one client (`<monitor_name>:<monitor_configuration_file>`). Each client has its own configuration file in `$HEALTH_CONF_DIR/monitors`. The configuration file specifies what and how to monitor.

A monitor usually consists of two files, a monitor script or binary and its configuration file. Each configuration file must contain the following configuration settings (common to all monitors):

- `MONITOR_NAME`

  Specifies the name of the monitor:

  `MONITOR_NAME=<monitor_name>`
- `MONITOR_TOOL`

  Specifies the tool monitoring the HP Operations management server:

  `MONITOR_TOOL=<monitor_tool_executed_by_health_daemon>`

This configuration file (with the absolute path) is used when registering the monitor. In addition to these configuration settings, other configuration settings that specify what and how to monitor can also be used.

The `RemoteAgent`, `ServerProcesses`, `MessageStorm`, and `ActiveMessages` monitors are available by default. However, it is also possible to set up a monitor according to your needs (for example, a monitor that would monitor particular parts of a particular management server). In this case, a monitor tool with the following usage must be provided:

```
-status
-issues
-syntax issues|status
```

The description of these options is as follows:

`-status`	Returns the status of one or more monitored sources. The returned status must match the syntax specified for the status of a specific monitored source in "Health Data" on page 262.
`-issues`	Returns all current issues (that is, all current problems) noticed by the monitor tool. The returned issues must match the syntax specified for the current issues in "Health Data" on page 262.
`-syntax issues|status`	Shows the syntax of all possible issues or statuses returned by the monitor tool.

Caution: To avoid unwanted timeouts, time needed for the monitor tool to get the status or issues of the monitored source needs to be as short as possible.

Examples of Health Monitoring Records

The following three examples show how to get the HP Operations management server health status record, the issue record, and the status record of all monitored sources:

Getting the HP Operations management server health status record

To get the HP Operations management server health status record, run the following command:

```
/opt/OV/bin/OpC/opchealth –status
```

A record similar to the following one appears:

```
MINOR:6:CRITICAL:1:
```

In this record, `MINOR` is the current status, `6` is the current rate, `CRITICAL` is the previous status, and `1` is the previous rate.

Getting the HP Operations management server issue record

To get the HP Operations management server issue record, run the following command:

```
/opt/OV/bin/OpC/opchealth –issues
```

A record similar to the following one appears:

```
AGENT_DOWN:4:nodeA:
SERVER_TRACING:5:
AGENT_TRACING:5:
```

```
NODE_DOWN:4:nodeX:
MESSAGE_STORM:3:nodeC:
PROCESS_DOWN:2:opccsad:
```

Getting the HP Operations status record of all monitored sources

To get the HP Operations management server status record of all monitored sources, run the following command:

```
/opt/OV/bin/OpC/opchealth -status -all
```

A record similar to the following one appears:

```
AGENT_DOWN:YES:7:
SERVER_TRACING:YES:5:
AGENT_TRACING:NO:10:
NODE_DOWN:NO:10:
MESSAGE_STORM:YES:3:
```

Converting the Health Status and Issue Text into Localized Text

To convert the health status and issue text into localized text, use the `opchealth2text.sh` tool that can be found at the following location:

```
/opt/OV/bin/OpC/utils/health/
```

The syntax of the `opchealth2text.sh` tool is the following:

```
opchealth2text.sh -health <item>
                  -status <item>
                  -issue <item>
                  -h|-?|-help
```

You can use the following options with the `opchealth2text.sh` tool:

`-health <item>`	Converts the general health status into localized text.
`-status <item>`	Converts the status of a monitored source into localized text.
`-issue <item>`	Converts an issue into localized text.
`-h|-?|-help`	Shows the usage.

Note: If you use the `-report` option with the `opchealth` tool, the health status and issue records are automatically converted into localized text.

Mapping of the health status and issue records into localized text is specified in the following configuration file:

```
$HEALTH_CONF_DIR/opchmi_messages.conf
```

The syntax of the configuration file is as follows:

```
STATUS:<monitored_source_name>:<status>:<rate>:
<catalog_set>:<catalog_message>:<default_C_message>

HEALTH:<current_status>:<previous_status>:
<catalog_set>:<catalog_message>:<default_C_message>

ISSUE:<issue_name>:<catalog_set>:<catalog_message>:
<default_C_message>
```

In these instances, the *<catalog_set>* and *<catalog_message>* parameters represent the message ID in a message catalog. If there is no such message in the catalog, the *<default_C_message>* parameter is shown. The *<status>* and *<rate>* parameters are optional. If these two parameters are not provided (that is, if they are empty), they are not matched when searching for the message ID.

Health Status Forwarding

The health monitoring daemon forwards health data to registered clients automatically. A client can be any product that is responsible for monitoring the status of the HP Operations management server or a third-party monitoring tool (see "HPOM Health Monitoring" on page 260). The management server itself can also be the client. The client in this case acts as a plugin for the health monitoring daemon.

The health monitoring daemon contains a configuration file with an internal list of the registered clients, `$HEALTH_CONF_DIR/HME_clients.conf`, in which each line represents one client (*<client_name>*:*<client_configuration_file>*). Each client has its own configuration file in `$HEALTH_CONF_DIR/clients`. The configuration file specifies when and what to forward to the clients.

The syntax of the client configuration file is as follows:

```
CLIENT_NAME=<client_name>

HEALTH_COMMAND=<command_for_forwarding_status> HEALTH_SEND=YES|NO
HEALTH_SEND_TYPE=ON_CHANGE|ON_TIME HEALTH_SEND_TIME=<time_between_sending_general_
status>

ISSUE_COMMAND=<command_for_forwarding_issues> ISSUES_SEND=YES|NO
ISSUES_SEND_TYPE=ON_CHANGE|ON_TIME ISSUES_SEND_TIME=<time_between_sending_issues>

ISSUES_SEND_WHAT=ALL|DIFF ISSUES_WHITELIST=<list_of_issue_names_that_can_only_be_
forwarded> ISSUES_BLACKLIST=<list_of_issue_names_that_are_not_ forwarded>

STATUS_COMMAND=<command_for_forwarding_status> STATUS_SEND=YES|NO
STATUS_SEND_TYPE=ON_CHANGE|ON_TIME STATUS_SEND_TIME=<time_between_sending_status>
STATUS_SEND_WHAT=ALL|DIFF
```

STATUS_WHITELIST=<*list_of_monitor_source_names_that_can_only_be_forwarded*>
STATUS_BLACKLIST=<*list_of_monitor_source_names_that_are_not_ forwarded*>

Note: The health data collected by the health monitoring daemon can also be used by third-party command line tools (by using the `opchealth` command line options such as `-health`, `-status`, and `-issues`).

Clients

The health monitoring daemon automatically forwards health data to one of the clients that must be registered. The following are the default clients:

- "LogFile Client" on page 268
- "opcmsg Client" on page 268
- "opcwall Client " on page 269
- "JET Client" on page 269

In addition to the default clients, you can also set up a client according to your needs (for example, a client that would communicate the HP Operations management server statuses and issues by sending SMS's, emails, and so on). To register such a client, you must prepare a configuration file containing client-specific configuration settings.

LogFile Client

The `LogFile` client writes all data to a log file. The data can be converted into localized text by using the `opchealth2txt.sh` tool.

The client-specific settings are as follows:

LOG_FILE=<*filename*>	Represents the file in which health data is stored.
CONVERT_TO_TEXT=YES	Converts the data into localized text. If you choose `NO`, the conversion does not take place.

opcmsg Client

The `opcmsg` client forwards health data to the HP Operations management server by using the `opcmsg` tool. The health data is converted into localized text and provided as the `msg_text` parameter.

The client configuration file contains the settings for the messages that are sent to the management server. These message settings (a message application, an object, severity, a group, and CMAs) can be customized.

The client-specific settings are as follows:

```
DEFAULT_GROUP=<message_group>
DEFAULT_APPLICATION=<application>
DEFAULT_OBJECT=<object>
DEFAULT_SEVERITY=<severity>
DEFAULT_CMA=<CMAs>

GROUP_<issue_or_status_name>=<message_group>
APPLICATION_<issue_or_status_name>=<application>
OBJECT_<issue_or_status_name>=<object>
SEVERITY_<issue_or_status_name>=<severity>
CMA_<issue_or_status_name>=<CMAs>
```

opcwall Client

The `opcwall` client forwards health data to the HP Operations management operators by using the `opcwall` tool. The health data is converted into localized text and provided as the `msg_text` parameter.

The client-specific setting is as follows:

```
WALL_USER=<HPOM_user>
```

JET Client

The `JET` client forwards health data to a JET environment (an HP specific environment) by using the `jetomuclient` tool. The health data is converted into localized text and provided as the `Message Text` attribute when issuing a JET alarm by using `jetomuclient`.

The alarm attributes (that is, a node type, an application, a group, an object, severity, a responsible user, an instruction, and CMAs) can be customized in the client configuration file.

The client-specific settings are as follows:

```
DEFAULT_NODE_TYPE=<node_type>
DEFAULT_GROUP=<message_group>
DEFAULT_APPLICATION=<application>
DEFAULT_OBJECT=<object>
DEFAULT_SEVERITY=<severity>
DEFAULT_CMA=<CMAs>
DEFAULT_USER=<responsible_user>
DEFAULT_INSTRUCTION=<instruction>

NODETYPE_<issue_or_status_name>=<node_type>
GROUP_<issue_or_status_name>=<message_group>
APPLICATION_<issue_or_status_name>=<application>
OBJECT_<issue_or_status_name>=<object>
SEVERITY_<issue_or_status_name>=<severity>
CMA_<issue_or_status_name>=<CMAs>
USER_<issue_or_status_name>=<responsible_user>
INSTRUCTION_<issue_or_status_name>=<instruction>
```

Agent Running and Reachable

The Agent Running and Reachable (ARR) component is a health monitor that can be controlled by the health monitoring daemon. All ARR events (for example, `NODE DOWN`, `AGENT DOWN`, and so on) are forwarded to the clients through the health monitoring daemon.

The ARR enables you to configure the following lists:

- List of all managed nodes that are controlled by the HP Operations management server (updated on a regular basis)

  For each managed node controlled by HPOM, the ARR first checks if it is accessible over the ping protocol. The following two scenarios are possible:

  - Managed node is accessible:

    If the managed node is accessible, the ARR checks the remote agent status by using the `opcragt` command. If there is no response, the remote agent is marked as `DOWN`. If there is a response, but the remote agent does not run properly, the ARR reports an agent failure.

  - Managed node is not accessible:

    If the managed node is not accessible, the ARR sends additional `ping` events. If there is still no response, the managed node is marked as `DOWN`.

- List of all manually added managed nodes that are not controlled by the HP Operations management server

  For each managed node not controlled by HPOM, the ARR checks only the ping status.

- List of excluded nodes

  Make sure that the `excluded_nodes` list contains all managed nodes for which you do not want to perform a status check.

To manage the ARR component, use the `opcarr` tool that you can find at the following location:

`/opt/OV/bin/OpC/`

The syntax of the `opcarr` tool is as follows:

```
opcarr -check [ <options> ]
        -report <options>
        -node <options>
        -group <options>
        -exclude <options>
        -pingopt <options>
        -reload
        -reindex
        -h|-?|-help
```

You can use the following options with the `opcarr` tool:

`-check`	Performs the following node status checks: • Status check of a selected node or all nodes. If you specify a node name, the selected node is checked. Otherwise, all nodes are checked. • Ping-only status check of all nodes or faulted nodes. • Agent-only status check of all nodes or faulted nodes.
`-report`	Shows the node or node group[a] status.
`-node`	Lists all monitored nodes, lists all node groups to which a specified node belongs, adds a node to a node list, or removes a node from a node list.
`-group`	• Lists all node groups. • Lists all members belonging to a selected node group. • Adds a new node group member or node group. • Removes a node group member or a node group.
`-exclude`	• Lists all excluded items. • Excludes a selected node or node group from the status check. • Excludes a selected node from the agent status check. • Removes a selected node check, node group check, or agent check from the excluded list.
`-pingopt`	• Lists ping options for a selected node or all nodes. • Sets ping options for a selected node or node group. You can specify either the full set of arguments for the ping command or only specific ping options, such as a package size, a timeout length, or the number of sent packages. • Clears ping options for a selected node or node group.
`-reload`	Updates a node list from the HP Operations management server.
`-reindex`	Reindexes node or node group relations and sets pings for all nodes.
`-h|-\?|-help`	Shows the usage.

For detailed information about all `opcarr` options, see the *opcarr* manual page.

[a]Nodes can be grouped into custom-defined groups consisting of group members—selected nodes, nodes that are members of an HP Operations management server group, or nodes that match a specific regular expression (for example, for segments, domains, and so on).

Note: The `arr.conf`, `arr_nodes.conf`, `arr_excluded_nodes.conf`, and `arr_ping_only_nodes.conf` configuration files can be found at the following location:

`/etc/opt/OV/share/conf/OpC/mgmt_sv/health/arr/`

In addition, the `arr.conf.sample` file with the description of the parameters that can be set in the `arr.conf` file can be found at this location.

ARR Event Logging

ARR event logging provides a standard way of recording important ARR events such as the status change of a node or a node group. The information about each event is stored in the following event log:

`/var/opt/OV/share/tmp/OpC/health/arr/arr.log`

Agent Health Check

The Health Check (HC) component is responsible for controlling a continuous message flow from managed nodes to management servers and for monitoring the HP Operations Agent health status that is received from managed nodes. The message flow starts with a scheduled action that is configured on a managed node to submit `Alive` messages for HPOM periodically. The messages are then diverted to the MSI and the `opchcd` management server process receives these messages and updates the HC statuses of the nodes. The `opchcd` process removes the `HC Alive` messages from the MSI after they are received so that they do not appear in the message browser.

To manage the HC component, use the `opchc.sh` tool that can be found at the following location:

`/opt/OV/bin/OpC/utils/hc/`

For detailed information about the `opchc.sh` tool usage and options, see the *opchc.sh* manual page.

Caution: Before configuring the HC component, make sure that the HP Operations Agent package (version 11.02.011 or higher) is installed on the HP Operations management server.

Enabling and Configuring the HC Component

The HC component can be installed and enabled during the installation of the HPOM software on the management server. For details, see the *HPOM Installation Guide for the Management Server*.

Note: If you decided not to enable the HC component during the management server installation, you can do it later by running the following command:

`/opt/OV/bin/OpC/utils/hc/opchc.sh -enable [SM]`

By running this command, you run the `opchcd` process with a certain initial configuration.

In this command, the `SM` argument is optional and you should use it if you want to enable the HC component integration with the HP Operations Agent self-monitoring feature. For details, see "Agent Health Check Integration with the HP Operations Agent Self-Monitoring Feature " on page 281.

Prerequisites for Enabling and Configuring the HC Component

Before enabling and configuring the HC component, make sure that the following prerequisites are met:

- Managed nodes are added or uploaded to the management server.
- The HP Operations Agent version 11.12 or higher is installed on managed nodes.

Configuring the HC Component

The following steps show which command or commands you should use for a particular action when configuring the HC component:

1. *Optional:* Enable the ARR component to monitor the status of all or selected managed nodes.

   For example, for all the managed nodes to be monitored by the ARR component, run the following commands:

   ```
   /opt/OV/bin/OpC/opcarr -reload
   /opt/OV/bin/OpC/opcarr -check
   /opt/OV/bin/OpC/opcarr -report node
   ```

   For detailed information about the ARR component, see "Agent Running and Reachable" on page 270.

2. Assign your remote node to a corresponding node group for monitoring:

   `HC-Interval:10m` for a 10 minute interval for HP Operations Agent 11.x

   `HC-Interval:1h` for a one hour interval for HP Operations Agent 11.x

   `HC-Interval:OA12-10m` for a 10 minute interval for HP Operations Agent 12.x

   `HC-Interval:OA12-1h` for a one hour interval for HP Operations Agent 11.x

   For example:

   ```
   opcnode -assign_node node_name=<remote_node_at_10m> \
   net_type=NETWORK_IP group_name=HC-Interval:10m
   opcnode -assign_node node_name=<remote_node_at_1h> \
   net_type=NETWORK_IP group_name=HC-Interval:1h
   opcnode -assign_node node_name=<OA12_node_at_10m> \
   net_type=NETWORK_IP group_name=HC-Interval:OA12-10m
   opcnode -assign_node node_name=<OA12_node_at_1h> \
   net_type=NETWORK_IP group_name=HC-Interval:OA12-1h
   ```

Caution: Every time the node assignment in the monitoring node group changes, make sure to run the `opchc.sh –compile` command to rebuild the HC configuration files.

3. Distribute the policies and instrumentation by running the following command:

```
opcragt -distrib -nodegrp HC-Interval:10m \
-nodegrp HC-Interval:1h \
-nodegrp HC-Interval:OA12-10m \
-nodegrp HC-Interval:OA12-1h
```

4. Build and verify the HC configuration files by running the following commands:

```
/opt/OV/bin/OpC/utils/hc/opchc.sh -compile
/opt/OV/bin/OpC/utils/hc/opchc.sh -verify
```

Note: The `opchc.sh -verify` command does not list the nodes from `HC-Interval:<interval>` node groups and HC configuration files. It lists only the nodes that are not monitored by the HC component.

For details about how to use the `opchc.sh` tool, see the *opchc.sh* manual page.

You can check the HC status of your monitored node or nodes by running the following command:

```
/opt/OV/bin/OpC/utils/hc/opchc.sh -report
```

Note: The `HC-Interval` time period should elapse before you run the HC status report. Keep in mind that after the initial HC policy distribution, it may take some time to receive first `HC Alive` messages and optionally `Self Monitoring` status messages. After the `HC-Interval` time period elapses, the HC status report is trustworthy.

Caution: If both the HC component and the ESF component are enabled, it is highly recommended to exclude the HC message group in the ESF flood gate configuration file (`ExcludeMsgGroup=HC`), so that all events with the `HealthCheck` message group are sent back to the MSI without matching any gate. Otherwise, the `HC Alive` messages coming from the monitored nodes might cause unwanted event storms and the HC status might be incorrect.

Disabling the HC Component

To disable the HC component, run the following command:

```
/opt/OV/bin/OpC/utils/hc/opchc.sh -disable
```

Caution: When disabling the HC component, you must first remove your nodes from corresponding `HC-Interval:<interval>` node groups and redistribute policies to them.

Otherwise, you may receive unwanted messages from such nodes even when the HC component is disabled.

Agent Health Check Configuration Files

When configuring the HC component, you can use the following two agent health check configuration files:

- "General Configuration File" on page 275
- "HC Interval Configuration File" on page 277

You can find the agent health check configuration files at the following location:

`/etc/opt/OV/share/conf/OpC/mgmt_sv/health/hc/`

General Configuration File

The general configuration file, `hc.conf`, contains statements that control the general behavior of HC modules (for example, turning on or turning off tracing, configuring used managed node or message group names, and so on).

Table 14 shows which parameters you can use to customize the general configuration file according to your preferences.

Table 14: General Configuration File Parameters

Parameter	Default Value	Description
`TRACE_LEVEL`	`0` (off)	Used to set the trace level (useful when troubleshooting the HC). Available trace levels are as follows: • `0` Tracing disabled • `1` Basic trace level • `2` Medium trace level • `3` Full trace level The trace file is located at: `/var/opt/OV/share/tmp/OpC/health/hc/log/hc.trc` This file is deleted each time the `opchcd` process is restarted.
`ALARM_MODE`	`file,msg`	Defines an HC alarm mode. If the alarm mode is `file`, alarm events are logged to the

General Configuration File Parameters, continued

Parameter	Default Value	Description
		file defined under ALARM_HIST. If the alarm mode is msg, alarm events are sent to the message browser as messages.
ALARM_HIST	/var/opt/OV/share/tmp/OpC/health/ /hc/log/hc.hist	Shows the location of HC alarming messages (that is, the HC history file).
HC_SELFMON_ENABLED	FALSE	Enables or disables the integration with the self-monitoring component.
HC_LOG_ARCHIVE	ON	Enables or disables the log archive mode.
HC_LOGSIZE_LIMIT	10	Maximum size of the hc.log file (in MB). If the size of the file exceeds the specified file size limit, it is renamed and moved to the HC_<*YYYYMMDD*> directory.
HC_KEEP_LOG	7	Specifies the number of days to keep HC_<*YYYYMMDD*>.tar.gz.
HCREPORT_LOG_ARCHIVE	ON	Enables or disables the log archive mode.
HCREPORT_KEEP_LOG	7	Specifies the number of days to keep HCREPORT_<*YYYYMMDD*>.tar.gz.
HCREPORT_INTERVAL	30	Specifies the frequency (in minutes) for generating the HC report log.
HCHIST_ARCHIVE	ON	Enables or disables the HC history archive mode.
HCHIST_KEEP_LOG	7	Specifies the number of days to keep <*ALARM_HIST*>.<*YYYYMMDD*>.tar.gz.

The following is an example of the configuration:

```
NODEGRP_PREF HC-Interval:
#NODEGRP_PREF C_

TRACE_LEVEL 0
ALARM_MODE file,msg
ALARM_CUST FALSE
ALARM_HIST /var/opt/OV/share/tmp/OpC/health/hc/log/hc.hist
```

To open the hc.conf file, run the following command:

```
/opt/OV/bin/OpC/utils/hc/opchc.sh -edit -conf
```

HC Interval Configuration File

Each HC interval configuration file, /etc/opt/OV/share/conf/OpC/mgmt_sv/health/hc/sla/<*interval*>.parms, consists of the parameters described in Table 15.

Table 15: HC Interval Configuration File Parameters

Parameter	Default Value	Description
INTERVAL	<*defined_interval_time*>	Time frame (in minutes) during which at least one HC alive message must arrive. Otherwise, the HP Operations Agent on the node is considered faulty. This parameter is not configurable. Therefore, make sure not to modify it.
DELAY	2	Additional buffer time for messages to be received on the HP Operations management server (in minutes)
RESEND	N/A	*Not added by default*: Resend time alarm message
ALARM_OBJ	HealthCheck	Object alarm message
ALARM_MSGGRP	OpC	Message group alarm message
ALARM_SEV	CRITICAL	Severity alarm message The following are the possible values: • NONE • UNKNOWN • NORMAL • WARNING • CRITICAL • MINOR • MAJOR
APP_SELFMON	SelfMon	Application for self-monitoring alarm message
THRESHOLD	<*half_of_INTERVAL_parameter_value*>	Defines how the opchcd process loads interval statuses when it is stopped and started again (in minutes):

HC Interval Configuration File Parameters, continued

Parameter	Default Value	Description
		• If the dead time of `opchcd` is longer than the time set for `THRESHOLD`, `opchcd` clears old statuses for the interval. • If the dead time of `opchcd` is shorter than the time set for `THRESHOLD`, `opchcd` loads old statuses for the interval.
`CMA_EVENT_SOURCE`	`MS_OVO`	Custom message attribute added to alarm messages
`CMA_EVENT_TYPE`	`hpoa`	Custom message attribute added to alarm messages

The interval is a time frame during which the HC component must detect a problem on a managed node and update its HC status. The problem can be one of the following:

- Message flow from the managed node is broken.
- When the self-monitoring component is enabled, some agent processes are not operating as expected on the managed node.

The interval is used by both the HP Operations Agent and the `opchcd` management server process. The HC policies on one or more managed nodes send `Alive` messages to the management server periodically. If more than `<interval>+<delay>` time period passes since the last message was received from the managed node, the HC component updates the `MsgFlowState` status from `UP` to `DOWN` for the corresponding node. In addition, it sends an alarming message to the message browser and logs this event into the `hc.hist` file.

Caution: The `DELAY` parameter must be introduced so that a potential delay in message delivery is also considered (for example, because of a slow network infrastructure).

The alarm message is sent once unless the `HC Alive` message arrives. If the `HC Alive` message arrives, the alarm state or the resend time is reset. If the managed node is permanently down, an alarm message is sent every `RESEND` minutes.

The following is an example of the configuration:

```
INTERVAL     10
RESEND       60
FREQUENCY    1

ALARM_MSGTXT Failure: The message flow is broken for the last 10 min from the
following node(s):
REALARM_MSGTXT HC up message received for the last 10min
ALARM_OBJ    HealthCheck
ALARM_MSGGRP OpC
```

```
APP_SELFMON SelfMon
ALARM_SEV CRITICAL
```

To open the HC interval configuration file, run the following command:

```
/opt/OV/bin/OpC/utils/hc/opchc.sh -edit -interval HC-Interval:<interval>
```

Agent Health Check Operator

The agent health check operator (that is, the HPOM user `HC_op` with the default password `HC_op`) is installed together with the HC component. This operator has the following preassigned elements:

- `HealthCheck` tool group
- `HealthCheck` message group in conjunction with all `HealthCheck` default node groups

All HPOM users that are supposed to receive the HC alarm message must have a responsibility matrix including the `HealthCheck` message group combined with the appropriate node groups. These node groups contain the real nodes and the pseudo node representing the customer.

Agent Health Check Tools

The `HealthCheck` tool group is installed together with the HC component. Table 16 shows the tools that can be used for managing the HC component.

Table 16: Tools for Managing the HC Component

Name/Label	Description
`Add HC Interval`	Creates HC interval objects.
`Compile HC Config`	Creates HC node configuration files for selected HC interval groups.
`Edit HC Config`	Edits the general HC configuration file on the HP Operations management server.
`Edit HC Interval`	Edits the HC interval parameters file on the HP Operations management server.
`HC Report`	Prints an HC report about the selected interval groups.
`HC Status`	Displays the HC program status on the HP Operations management server.
`List HC Intervals`	Lists all HC intervals.
`Manual Test`	Sends a test active message for a node.

Tools for Managing the HC Component, continued

Name/Label	Description
Outage Auto	Sends an outage auto message (that is, sets WantState to DOWN(A)) for a node to prevent more critical messages and WantState should go to UP after first Alive message is received.
Outage End	Sends an outage end message (that is, sets WantState back to UP) for a node.
Outage Start	Sends outage start message (that is, sets WantState to DOWN(P)) for a node.
Reinit HC	Reinitializes the HC program on the HP Operations management server.
Remove HC Interval	Removes HC interval objects.
Start HC	Starts the HC program on the HP Operations management server.
Stop HC	Stops the HC program on the HP Operations management server.
Verify Nodes	Verifies if HPOM nodes are monitored by the HC component.

Agent Health Check Logging and Tracing

When troubleshooting, you can analyze the files inside the following directory to find out when and where new events occurred:

/var/opt/OV/share/tmp/OpC/health/hc/log/

These files are as follows:

hc.hist	File that stores all alarming events detected by the HC component. These events are sent to the message browser as HPOM messages (this behavior is set by default in the hc.conf file).
hc.log	Log file that keeps all internal warnings and/or errors of the HC component. The hc.log file is cleared after the opchcd process is restarted.
hc.trc	Trace file for the opchcd process and the opchc.sh tool. You can enable HC tracing by using the HC_TRACE parameter in the hc.conf file. The hc.trc file is cleared after HC tracing is disabled and the opchcd process is reinitialized or restarted.

Agent Health Check Integration with the ARR Component

Because the ARR component may contain a list of all the managed nodes that are controlled by the HP Operations management server, it enables you to check the status of each of these managed

nodes. In fact, the HC compares its status with the ARR status for each managed node that is monitored by the HC. For example, if the HC detects that node A does not send any `HC Alive` messages during the defined interval of time and if the ARR status is also `DOWN`, the HC does not generate any alarm message because the ARR is already informed about the problem. By not generating any alarm message, reporting the same problem two times is avoided.

Note: Although the HC component can run without the ARR component being enabled, it is recommended that you enable it.

For detailed information about the ARR component, see "Agent Running and Reachable" on page 270.

Agent Health Check Integration with the HP Operations Agent Self-Monitoring Feature

In addition to the agent self-monitoring feature introduced with HP Operations Agent 11.x, a new agent self-monitoring feature is available with HP Operations Agent 12.x. You can integrate the agent health check feature with the agent self-monitoring feature.

After the initial installation of the HC component, you can enable the integration with the agent self-monitoring feature. To do this, run the following command:

```
/opt/OV/bin/OpC/utils/hc/opchc.sh -enable SM
```

Caution: After you enable the integration with the agent self-monitoring feature, make sure to follow the steps described in "Configuring the HC Component" on page 273. By doing so, you will reconfigure the HC component and redistribute the needed policies to the managed nodes.

When you install the HP Operations Agent deployment package on the HP Operations management server, the `Self Monitoring` policies and instrumentation are uploaded. These policies and instrumentation are used by the HC component for smooth functioning of the HP Operations Agent, as their main functions are to monitor the health of the HP Operations Agent and to help you make sure that necessary agent processes respond and are not stopped.

Enabling the integration with the agent self-monitoring feature results in additional `Self Monitoring` policies being assigned and distributed to one or more managed nodes. For HP Operations Agent 11.x, the `SelfMonVerifyAll2HC:<interval>` policy is responsible for the integration and it regularly sends health status messages of the HP Operations Agent to the HC component on the management server.

For HP Operations Agent 12.x, `nodeinfo` policies cause the agent to regularly send health status messages to the HC component on the management server.

The HC status report also shows the self-monitoring status, for example:

```
# Status of Node Group '10m'
# ------------------------------------------------------------
```

```
# NodeName          WantState  MsgFlowState  SelfMonState
#           LastActMsgTime                    MsgFlowAlarmTime
# ------------------------------------------------------------
<managed_node>           UP           UP       UP
            Fri Apr 25 13:12:01 2014           n/a
```

If you want to disable the integration with the agent self-monitoring feature, run the following command:

```
/opt/OV/bin/OpC/utils/hc/opchc.sh -disable SM
```

Caution: After you disable the integration with the agent self-monitoring feature, make sure to follow steps 3 and 4 described in "Configuring the HC Component" on page 273.

For detailed information about the agent self-monitoring feature, see the HP Operations Agent documentation.

HPOM Agent Health Check changes for Operations Agent 12.x

This section describes how a new agent self-monitoring feature introduced with HP Operations Agent 12.x is integrated with the HPOM health check feature. If you configure this integration, your environment acquires the following capabilities:

- The agent regularly checks the health of the subagents and sends any change in the health status to the management server.
- The health status can also be viewed through a web browser by using the dashboard view on the management server.
- The agent sends an `alive` message at regular intervals.

Note: For detailed information about the agent self-monitoring feature, see the HP Operations Agent 12.x documentation.

The integration of the HPOM health check feature and the new HP Operations Agent self-monitoring feature includes the following changes:

- The following lines are added to the `bbc_inst_defaults` sample file, `bbc_inst_defaults.sampl`:

  ```
  ; Enable Agent Self-Monitoring on OA 12.x managed nodes.
  [agent.health]
  OPC_SELFMON_ENABLE=TRUE
  ```

  The location of the `bbc_inst_defaults.sampl` file is as follows:

  `/etc/opt/OV/share/conf/OpC/mgmt_sv/`

  The default self-monitoring interval and the management server are used as the dashboard view target for new management server-based installations.

Caution: These changes take effect when you copy the `bbc_inst_defaults.sampl` file to the `bbc_inst_defaults` file, or add it to the existing `bbc_inst_defaults` file.

- The `opchcd` MSI program listens to the health check messages and uses them in the status. For example, the output of the `opchc.sh –report` command could look like this:

```
# Status of Node Group 'OA12-10m'.
# ---------------------------------------------------------
# NodeName     WantState MsgFlowState SelfMonState
#      LastActMsgTime MsgFlowAlarmTime
# ---------------------------------------------------------
dhcp-agent.hp.com UP           UP      Warning
Fri Sep 19 14:09:12 2014 n/a
…
# Status of Node Group '10m'.
# ---------------------------------------------------------
# NodeName     WantState MsgFlowState SelfMonState
#      LastActMsgTime MsgFlowAlarmTime
# ---------------------------------------------------------

tmcentos.hp.com   UP           UP      DOWN: opcle
Fri Sep 19 14:09:14 2014 n/a
```

  In this example, `dhcp-agent` is HP Operations Agent 12.x and `SelfMonState` is `Warning` because an HP Operations Agent 12.x health message with severity `Warning` was received. The HP Operations Agent 12.x health message also appears in the message browser because it has additional information about the issue. `tmcentos` is a managed node with HP Operations Agent 11.x where the old HPOM`SelfMon` policies and the corresponding status are assigned.

- Two new node groups, `HC-Interval:OA12-10m` and `HC-Interval:OA12-1h`, with corresponding policy groups are added. Instead of assigning a managed node with HP Operations Agent 12.x to the old `HC-Interval:10m` node group, you assign it to the `HC-Interval:OA12-10m` node group. However, the agents with the older agent versions should still be assigned to the `HC-Interval:10m` node group. Instead of using a scheduled action and the agent `SelfMon` policies, the `nodeinfo` policies, which enable the new HP Operations Agent 12.x self-monitoring feature, are assigned.
- Because HP Operations Agent 12.x sends only an alive message, if no regular message is sent within the defined interval, the `opchcd` MSI program handles any message from a managed node as an alive message (excluding proxy messages). This also includes old managed nodes that have old HPOM health check policies.
- When the agent is stopped manually, HP Operations Agent 12.x sends a `Critical` heartbeat message. The `opchcd` MSI program replaces this message with a `Warning` message indicating that the agent is stopped on a specific node and the `MsgFlowState` is set to `Down`.
- When the health check is enabled, the new health check configuration (heartbeat part) is uploaded

together with the old health check node groups, policies, and so on. This new configuration is also uploaded when the following command is run again:

```
/opt/OV/bin/OpC/utils/hc/opchc.sh –enable
```

- When the agent self-monitoring feature is enabled (that is, when the `opchc.sh –enable SM` command is run), the new self-monitoring configuration (enabling the HP Operations Agent 12.x self-monitoring feature) is uploaded. This assigns the self-monitoring `nodeinfo` policy to the HP Operations Agent 12.x policy groups and thus to the nodes in the `HC-Interval:OA12-10m` and `HC-Interval:OA12-1h` node groups. This approach is similar to the one for older agents where the self-monitoring policies are assigned to existing policy groups.

To include new HP Operations Agents 12.x in the HPOM health check, follow these steps:

1. Re-enable the health check feature in order to upload new node groups, policy groups, and policies by running the following command:

   ```
   /opt/OV/bin/OpC/utils/hc/opchc.sh –enable
   ```

   Note: Perform this step only once after the upgrade to HPOM 9.21.

2. Re-enable the agent self-monitoring feature in order to upload new agent self-monitoring `nodeinfo` policies:

   ```
   /opt/OV/bin/OpC/utils/hc/opchc.sh –enable SM
   ```

   Note: Perform this step only once after the upgrade to HPOM 9.21.

3. Assign the node to the `HC-Interval:OA12-10m` or `HC-Interval:OA12-1h` node group.
4. Distribute the configuration to the managed node(s). For example, run the following command:

   ```
   /opt/OV/bin/OpC/opcragt -distrib -templates -nodegrp HC-Interval:OA12-10m
   ```

5. Recompile the health check configuration by running the following command:

   ```
   /opt/OV/bin/OpC/utils/hc/opchc.sh -compile
   ```

For example, if you now run the `opchc.sh -report` command, it shows the new HP Operations Agent 12.x intervals in addition to the old intervals:

```
/opt/OV/bin/OpC/utils/hc/opchc.sh –report

# ============================================================
     # Status of HC Interval 'OA12-1h'.
     # ============================================================
     Last check: Thu Dec 11 10:52:10 2014

     # ============================================================
     # Status of HC Interval 'OA12-10m'.
     # ============================================================
     Last check: Thu Dec 11 10:54:26 2014
```

```
# Status of Node Group 'OA12-10m'.
# ------------------------------------------------------------
# NodeName         WantState  MsgFlowState  SelfMonState
#           LastActMsgTime                    MsgFlowAlarmTime
# ------------------------------------------------------------
agent1.hp.com        UP         UP          Critical
           Thu Dec 11 10:54:11 2014           n/a

tmcentos.hp.com      UP         UP             UP
           Thu Dec 11 10:53:57 2014         n/a
```

Part III: User Interfaces

HP Operations Manager (HPOM) provides two user interfaces. The Java-based graphical user interface (GUI) is an easy-to-use interface for HPOM operators. Service Navigator is a component of the Java GUI.

The administration user interface is used by HPOM administrators to administer the HPOM managed environment.

For detailed information about the HPOM user interfaces, see:

- "HPOM Java GUI" on page 287
- "HPOM Service Navigator" on page 341
- "HPOM Administration UI " on page 403

Chapter 8: HPOM Java GUI

In this Chapter

The information in this chapter describes the Java-based graphical user interface (Java GUI) for the HP Operations Manager (HPOM) operator. In this chapter, you can find information about the following topics:

For more detailed information about installation requirements and instructions, see the *HPOM Installation Guide for the Management Server*.

Java GUI Overview

This section provides an overview of how the HPOM Java-based operator GUI handles the message browsers. It also describes how windows are refreshed and users are viewed. For more detailed information about the HPOM Java-based operator GUI functionality, see the *HPOM Java GUI Operator's Guide*.

The HP Operations Manager Java-GUI provides HPOM operators with a graphical user interface that is extremely easy to use. The Java GUI can runs on any platform where the Java Runtime Environment

(JRE) is installed. HPOM enables operators to connect to HPOM running on a variety of platforms. The Java GUI provides the following high-level features:

- Refreshing windows:

  The Java GUI automatically updates the status of nodes, message groups, messages, and services if applicable at a preset interval. In the Java GUI, you can reconfigure this refresh interval. When you press the `[Acknowledge]` button in the `Message Properties` window, the node coloring in the object pane is not immediately updated. However, you can manually refresh the node coloring by pressing the `Refresh` toolbar button or by selecting the menu `View: Refresh`. Or can wait until the next automatic refresh is completed.

- Viewing users:

  The Java GUI does not create an entry in the database table `opc_op_runtime` for currently active HPOM users. As a result, the reports listing `Unmonitored` and `Working HPOM Users` do not include Java GUI users.

Message Browsers

The Java GUI message browser provides the following features and functionality:

- Customizing message columns:

  The HPOM Java GUI lets you resize, move, hide, and change the order of the columns in the message browsers. The Java GUI also lets you sort messages according to message attributes. For example, you can sort messages by date and Time, by node, or by application.

- Setting a filter search:

  By default, the Message Text filter search in the message browser is case sensitive.

  To change the default settings, make sure to update the `OPC_CASE_SENSITIVE_SEARCH` server configuration variable to `FALSE` as follows:

  ```
  ovconfchg -ovrg server -ns opc -set OPC_CASE_SENSITIVE_SEARCH FALSE
  ```

- Displaying messages:

  In the Java GUI, you can choose between displaying all messages or only the most recent messages. You can define the number of messages displayed.

- Setting flags:

  Java GUI does not immediately update the flags in the SUIAONE columns, which indicate message severity, ownership, action availability and status, and so on. In exceptional circumstances, it is possible for an operator-initiated action to complete before the status in the browser is set to started.

- Acknowledging messages:

  To acknowledge messages based on severity, open a View Message Browser, choose a level of severity as the filtering criteria, and acknowledge all messages in the current view. Alternatively,

click the Severity column in the browser to sort the messages by severity, select the messages with level of severity you want, and acknowledge all messages in the current view.

- Owning messages:

  The Java GUI enables you to own only selected messages. If you want to own *all* messages in a message browser, change the preferences settings so the browser displays all messages, then select and own them all.

Startup Options

This section describes the configuration options evaluated by the Java GUI when it is started with the ito_op startup script. When the Java GUI starts, it reads environment options first, then evaluates any command-line options passed with the startup script, and finally considers the contents of the itooprc file.

Starting the Java GUI with the ito_op Script

To start the Java GUI with the ito_op script, enter the following command:

```
# /opt/OV/www/htdocs/ito_op/ito_op &
```

For more information about the options you can set in the ito_op script and how you can use them to control the look, feel, layout and content of the Java GUI, see the following tables:

- Table 17: Communication, security, and display.
- Table 18: Layout, content, and workspace.

Table 17 shows the options evaluated by the Java GUI in the startup scripts. The options include: settings for communication (ports, proxies, and servers), security, passwords, and so on.

Table 17: Startup Script Options Evaluated by the Java GUI

Option	Format	Default Value	Description
apisid	<*string*>	OV_JGUI_API	Sets a session ID for the particular Java GUI instance at its startup.
bbc.http:proxy	<*string*>	“”	Configures a proxy server for HTTPS-based communication.
colored_ message_lines	yes|no	no	Decides whether whole messages or only the severity column are colored in the message browser.

Startup Script Options Evaluated by the Java GUI, continued

Option	Format	Default Value	Description
def_browser	<*filename*>	""	Specifies the path to the default web browser on localhost.
def_look_and_feel	<*string*>	Windows: com.sun.java.swing.plaf.motif.MotifLookAndFeel	Defines the appearance of the Java GUI.
display	<*host.domain*>:0	<*localhost*>:0	Specifies the hostname to which the X application redirects the display.
initial_node	<*string*>	<*localhost*>	Defines hostname of the HPOM management server to which the Java GUI connects.
locale	<*lang_territory*>		Sets locale name.
maxheap	<*number*>	256MB	Used for specifying the maximum heap memory size for the Java Virtual Machine. This heap memory is used for literal strings and class loader. You can specify a greater value according to your environment needs.
maxperm	<*number*>	128MB	Specifies the maximum permanent memory size for the Java Virtual Machine. This kind of memory is used for dynamically created objects.
max_limited_messages	<*int*>	50	Specifies maximum number of messages displayed in a browser.
nosec	true|false	false	Starts the SSL Secure Java GUI in standard mode without SSL functionality.
passwd	<*string*>	""	Defines password of the HPOM operator used for logon.

Startup Script Options Evaluated by the Java GUI, continued

Option	Format	Default Value	Description
port	See "Setting the Port for Non-Secure Socket Communication" on page 294.	2531	Sets the port number the Java GUI uses when connecting to the HP Operations management server.
refresh_ interval	<int> (seconds)	30	Defines the frequency with which the contents of the message browser are refreshed.
server	<string>	<localhost>	Specifies the hostname of the HPOM management server to which the Java GUI will connect.
title_suffix	<string>	""	Displays the string next to the title in the main window.
trace	true|false	false	Enables the appearance of tracing messages in the terminal.
user	<string>	""	Specifies the name of the HPOM operator used to log on.

Table 18 shows the options and attributes that you can use to control the layout and content of the Java GUI in the startup scripts. The options include: look and feel, layout, the workspace, browser types, and so on.

Table 18: Attributes Controlling the Layout and Content of the Java GUI

Name	Value	Default	Overrides	Details
gui.dftllayout	boolean	false		Controls the base layout. [a]
gui.objectpane	boolean			Shows or hides Object Pane.

[a]The attribute controls the base layout of the Java GUI to which the new objects, controlled by other attributes, will be added. If set to false (default), layout is blank. Additionally, if the message browser is opened on the browser pane, it will take 100% of the GUI (the horizontal splitter, dividing the workspace pane and browser pane will be on the top-most position). If a service graph is opened in the workspace, then the GUI is shared equally between the workspace and browser pane. If set to "true", the Java GUI is opened as today: if session-specific settings are found they are used, otherwise the global defaults are used.

Attributes Controlling the Layout and Content of the Java GUI, continued

Name	Value	Default	Overrides	Details
gui.shortcutbar	boolean			Show or hide Shortcut Bar.
gui.workspace	<name>	Default names as generated for new workspaces.		Create new workspaces.
gui.msgbrw.type	active | history | pending	active		Opens a browser with active, history, or pending messages.
gui.msgbrw.workspace	<name>	Default – first - workspace		Opens a browser in specified workspace.
gui.msgbrw.brwpane	<boolean>		gui.msgbrw. workspace	Opens a browser in browser pane.
gui.msgbrw.filter. name	<name>		gui.msgbrw. filter.<ANY>	A saved filter name overrides all filter attribute values.
gui.msgbrw.filter. nodes	<name_list>			
gui.msgbrw.filter. services	<name_list>			
gui.msgbrw.filter.apps	<name_list>			
gui.msgbrw.filter. msggrps	<name_list>			
gui.msgbrw.filter. objects	<name_list>			
gui.msgbrw.filter. msgtext	<string>			
gui.msgbrw.filter.time. start	<date/time>	today 0:00:00		date / time format as specified by the

Attributes Controlling the Layout and Content of the Java GUI, continued

Name	Value	Default	Overrides	Details
				system locale setting
gui.msgbrw.filter.time.end	<date/time>	today 23:59:59		date / time format as specified by the system locale setting
gui.msgbrw.filter.time.relative.start	<string>			the relative time syntax [+|-]<int> [d|h|m|s]
gui.msgbrw.filter.time.relative.end	<string>			the relative time syntax [+|-]<int> [d|h|m|s]
gui.msgbrw.filter.owned	not | me | others			
gui.msgbrw.filter.severity	<severity_list> enum {unknown, normal, warning minor, major, critical}			
gui.svcgraph.name	<service_name>	top level service		All services assigned to operator.
gui.svcgraph.calcid	<calc_id> (0 | 1)	0		service status calculation id
gui.svcgraph.workspace	<name>	Default (first) workspace		opens a graph in specified workspace.
gui.svcmap.name	<service_name>	top level service		All services assigned to operator
gui.svcmap.calcid	<calc_id> (0 | 1)	0		service status

Attributes Controlling the Layout and Content of the Java GUI, continued

Name	Value	Default	Overrides	Details
				calculation id
gui.svcmap. workspace	<name>	Default (first) workspace		opens a map in specified workspace.

Setting the Port for Non-Secure Socket Communication

To set the port through which the connection between the Java GUI and the HP Operations management server will be established, choose one of the following methods:

- In the `itooprc` resource file, define the following variable:

  `port=<port_number>`

- In the `ito_op` (`ito_op.bat` on Windows) start-up script, add one of the following sets of configuration parameters:
- `ito_op ... -port <port_number> ...`
- `ito_op <hostname>:<port_number> ...`
- `ito_op ... -server <hostname>:<port_number> ...`
- In the log-on dialog from the Management Server field, define the following parameter:

  `<hostname>:<port_number>`

Time-Zone Settings in the ito_op.bat File

The Java GUI displays time-related information according to the format and settings defined by the local time zone of the client. If the Java GUI and the HP Operations management server are located in different time zones, you can force the Java GUI to use the time zone of the management server by setting the `-Duser.timezone=<time_zone>` switch in the `ito_op.bat` file.

For example, to use the time zone `Australia/Sydney`, add the text `-Duser.timezone=Australia/Sydney` to the `ito_op.bat` file (example extract):

```
:: Starting JavaGUI
for %%p in (true TRUE on ON yes YES) do if "%%p"=="%TRACE%" echo on
for %%p in (true TRUE on ON yes YES) do if "%%p"=="%PLUGIN%" goto :PLUGIN
%START% .\j2re1.4.2\bin\%JAVA% -Duser.timezone=Australia/Sydney -Xmx128m
com.hp.ov.it.ui.OvEmbApplet initial_node=%ITOSERVER% user=%USER% passwd=%PASSWD%
trace=%TRACE% display=%DISPLAY% locale=%LOCALE%
max_limited_messages=%MAX_LIMITED_MESSAGES% refresh_interval=%REFRESH_INTERVAL%
```

```
apiport=%APIPORT% apisid=%APISID% https=%HTTPS% %BBCPARM%
goto END
```

Valid time zones are listed in the directory *<JRE_HOME>*\lib\zi, for example GMT, Asia/Singapore, or Europe/Warsaw. If you specify an invalid time zone, GMT is used.

Resource Files

The Java GUI resource file itooprc resides in the home directory of the user who starts the Java GUI and is used to store operator preferences. Each defined option must be listed in a separate line and followed by its parameter. The itooprc file is updated automatically after clicking the OK button in the Preferences dialog.

Caution: The itooprc file should be edited only by experienced administrators or operators.

Table 19 lists the configuration options that you can define in the Java GUI resource file, shows the required format, and briefly describes the result.

Table 19: itooprc Options and Parameters

Option	Format	Description
apiport	*<positive_integer>*	Port number used by the Java API to connect to the Java GUI.
apisid	*<string>*	Sets a session ID for the particular Java GUI instance at its startup.
apply_message_view_filter_to_status_summary	on|off|true|false|yes|no	Sets the status summary area to show the number and the severity of messages in the message browser tab based on the applied message view filter.
bbc.http:proxy	*<string>*	Configures a proxy server for HTTPS-based communication.
cb_port	*<positive_integer>*	Port number used by the Communication Broker (ovbbccb). The default port is 383.
chg_source_to_source_pol	yes|no	If enabled, it allows you to change the Java GUI message browser column name from Source to Source Policy.

itooprc Options and Parameters, continued

Option	Format	Description
		The default value is no. NOTE: Because of performance reasons, the Source column cannot display the condition parameter for all messages in the message browser.
colored_message_lines	on|off|true|false|yes|no	Enables you to color the entire message row in the message browser with the severity color of that message.
custom_status_id	<*non-negative_integer*>	Sets the user selected service status calculation view (0 is for Overall and 1 is for Operational).
def_help_url	<*url*>	Path to the help pages on the management server.
def_look_and_feel	<*look_and_feel*>	Defines the appearance of Java GUI: Metal, Motif, or Windows.
def_mvf_operator	equals|greater than|greater than or equals|less than|less than or equals|begins with|ends with|contains	Used for customizing the default message view filter operator. The default value is contains. NOTE: The columns that have numerical values do not have the begins with, ends with, or contains operator.
default_browser	<*path_to_browser*>	Path to the web browser on a local host.
display	<*hostname*>	Hostname of the exported display where X applications will be launched.
export_all_windows_msgs	yes|no	If enabled, all selected messages from all open message browsers can be exported to a file. The default value is no. NOTE: To select the messages, press and hold down the CTRL key and click

itooprc Options and Parameters, continued

Option	Format	Description
		the messages you want to export.
frmt_uncomplete_html	on|off|true|false|yes|no	If enabled, incomplete HTML code from a message instruction interface is formatted.
global_settings_poll_interval	<*non-negative_integer*>	Determines how frequently the Java GUI checks for changes to the global property files. Default is five minutes.
horizontal_zoom	<*positive_integer*>	Determines the default service graph horizontal zoom.
https	on|off|true|false|yes|no	Tells the Java GUI to use a secure connection.
https_conn_timeout	<*positive_integer*>	Defines the timeout period (in seconds) for HTTPS requests and replies.
https_fallback	<*to_socket*|*to_bbc*|*none*>	Configures the fallback mechanism responsible for the Java GUI client and HP Operations management server connection used in a secure communication type. This configuration cannot be specified as a global setting.
https_port	<*positive_integer*>	Specifies the port number (for example, 383) for secure connections.
initial_node	<*hostname*|*ip*>	Hostname of the HPOM management server to which the Java GUI will connect.
install_dir	<*path*>	Defines the location in which the HPOM was installed. *For HP internal use only*.
javagui_font_size	<*positive_integer*>	Sets the size of the application fonts (except in service graphs).
locale	<*locale_setting*>	Presets the locale name.
lcore_defaults	on|off|true|false|yes|no	Use the HPOM agent default locations.

itooprc Options and Parameters, continued

Option	Format	Description
`max_limited_history_messages`	<*positive_integer*>	Determines how many history messages to display in the message browsers.
`max_limited_messages`	<*positive_integer*>	Determines how many messages to display in the message browsers.
`message_notification_dlg`	`on|off|true|false|yes|no`	Shows a warning dialog when a message event occurs.
`message_notification_dlg_app`	`on|off|true|false|yes|no`	Starts a local application that will be executed when a message event occurs.
`message_notification_dlg_app_path`	<*path*>	Path to the local application that will be started when a message event occurs.
`message_notification_show_all`	`on|off|true|false|yes|no`	Sends event notification either for the first message to arrive or for every new message.
`noapp`	`true|false`	If set to `true`, both the Start and Start Customized... actions are disabled regardless of the number of applications that is assigned to the user. In addition, the Tailored set of Tools option is also disabled. The default value is `false`.
`nosec`	`on|off|true|false|yes|no`	Starts the SSL Secure Java GUI in standard mode without SSL functionality.
`number_of_frequently_used_tools`	<*non-negative_integer*>	The number of most frequently used tools that are displayed above the separator line in the pop-up menus from which you start the tools. The default value is 5, but you can set it in the range of 0 through 15.

itooprc Options and Parameters, continued

Option	Format	Description
passwd	<*password*>	Password of the HPOM operator used for logon.
popup_history_ interval	<*positive_integer*>	Used for setting the maximum time (in days) for history messages to be viewed in the message filter (when in the limited mode).
port	<*positive_integer*>	Port number the Java GUI uses to connect to the management server.
prompt_for_activate	on|off|true|false|yes|no	For HP internal use only.
reconnect_interval	<*positive_integer*>	Time (in seconds) the Java GUI allocates for reconnecting to the management server. Enabled also for the global itooprc resource file if the OPC_JGUI_ RECONNECT_FROM_GLOB_SETT server configuration variable is set to TRUE.
reconnect_timeout	<*positive_integer*>	Time (in seconds) after which the Java GUI stops reconnecting to an unreachable management server. Enabled also for the global itooprc resource file if the OPC_JGUI_ RECONNECT_FROM_GLOB_SETT server configuration variable is set to TRUE.
refresh_interval	<*positive_integer*>	Determines how frequently the Java GUI refreshes automatically. Default is 30 seconds.
secure_port	<*positive_integer*>	Port number the Secure Java GUI uses to connect to the management server.
select_only_managed_ nodes	on|off|true|false|yes|no	Used for selecting nodes. The default value is false. If enabled, only the regular node or

itooprc Options and Parameters, continued

Option	Format	Description
		nodes are selected (if no regular nodes are found, the external node or nodes are selected). If disabled or omitted (that is, not set), all nodes are selected.
service_graph_font_size	<*positive_integer*>	Sets the default service graph font size.
service_icon_zoom	<*integer_between_1_and_100*>	Used for customizing the size of service icons and severity status indicators. Can be set in the range of 1 through 100.
severity_label	text|both|icon	Determines whether the message browsers display icons, text, or both in the severity column.
shortcut_tree_icon_width	<*positive_integer*>	Controls the size (in pixels) of icons. Default is 32 pixels.
show_at_severity	0|1|2|3|4|5	Defines the severity of the message for which event notification takes place: 0 = Unknown 1 = Normal 2 = Warning 3 = Minor 4 = Major 5 = Critical
show_operator_as_services_root	on|off|true|false|yes|no	If enabled, the Services root node in the service graph is no longer named Services, but after the operator to whom the service was assigned. IMPORTANT: If the service configuration file does not contain the operator name inside the <Operator> tag, the Java GUI does not replace the

itooprc Options and Parameters, continued

Option	Format	Description
		Services root node with the operator name.
`show_svc_label_in_msgs`	`on|off|true|false|yes|no`	Enables the service label column in the message browser. If you update the service label column manually, make sure that you do it before the startup.
`show_comm_status_dlg`	`yes|no`	Java GUI clients check for changes to the global property files in the shared location. If a change is detected, the HPOM Communication Status dialog box displays a message that informs the operator of the changes and requests a restart of the Java GUI. This parameter enables or disables the HPOM Communication Status dialog box from displaying. The default value is `yes`. This is useful if you want to disable this behavior in an environment with many clients logged on with the same user name at the same time. This way, when one of these clients reloads the Java GUI configuration, the HPOM Communication Status dialog boxes are refrained from disturbing all the other clients.
`socket_fallback`	*<to_bbc|none>*	Configures the fallback mechanism responsible for the Java GUI client and the HP Operations management server connection by using a non-secure communication type. This configuration cannot be specified as a global setting.

itooprc Options and Parameters, continued

Option	Format	Description
sort_owned_severity	on|off|true|false|yes|no	Filters owned messages on the message browser by severity. The default value is no.
stay_on_top	on|off|true|false|yes|no	Enables the Java UI windows (main and detached) to stay on top of other windows. The default value is no.
subproduct	<*subproduct_string*>	For HP internal use only.
tailored_ applications_start	on|off|true|false|yes|no	Enables you to include only applications related to the selected message in the pop-up menus.
title_suffix	<*title*>	Displays the string next to the title in the main window.
trace	on|off|true|false|yes|no	Enables display of tracing messages in the terminal.
user	<*username*>	HPOM operator name used for logon.
vertical_zoom	<*positive_integer*>	Determines the default service graph vertical zoom.
web_browser_type	external|activex	Type of web browser to use in the workspace pane: • external Selects an external web browser. • activex Selects the Internet Explorer ActiveX control.
web_browser_html_ appl_result	true|false	*Internal* true= Enables the HTML output of the application. false= Disables the HTML output of the application.

Cockpit View

The HPOM cockpit view is a web-based interface that displays the state of the environment monitored by HPOM. The cockpit view helps users to quickly assess the current health of the monitored environment and its readiness to support the business. For a detailed description of the cockpit view, see the *HPOM Java GUI Operator's Guide*. Note that you can configure as many cockpit views as you need. The information in this section describes and explains the following aspects of the cockpit view:

Configuring the Cockpit View

To configure the cockpit view, perform the following steps:

1. Configure a layout configuration file for each cockpit view that you want to display.

   For more information about the contents of a cockpit view's layout- configuration file, see "Layout Configuration Files" on page 304.

2. Validate your layout configuration files against the Document Type Definition (DTD).

   For more information about a validating a cockpit view's layout configuration, see "Valid Layout Configuration Files" on page 318.

3. On the management server, store your layout configuration files in the following directory:

   `/opt/OV/www/htdocs/ito_op/assets/xml`

4. Start a cockpit view on the client system, type the following URL:

   - Standard connection:

     ```
     # http://<management_server>:8081/OvCgi/ito_op_applet_
     cgi.ovpl?cockpitview=true&view=<layout>
     ```

   - Secure connection:

     ```
     # https://<management_server>:8444/OvCgi/ito_op_applet_
     cgi.ovpl?cockpitview=true&view=<layout>
     ```

   <management_server> is the hostname of your management server, and *<layout>* is the name of the layout configuration file. (Omit the `.xml` file type extension.)

   For example, the following URL starts the sample cockpit view provided by HP:

   ```
   # http://<management_server>:8081/OvCgi/ito_op_applet_cgi.ovpl?cockpitview=true
   ```

Layout Configuration Files

The layout configuration files (*<layout>.xml*) determine the colors, layout, and contents of the indicator panel of a cockpit view. On the management server, layout configuration files are located in the following directory:

`/opt/OV/www/htdocs/ito_op/assets/xml/<layout>.xml`

Tip: The `xml` directory contains a sample layout configuration (`layout_simple.xml`) which you can use to get started. If you want to use the sample layout, do not edit the sample file itself. First, make a copy of the sample file, and edit the copy. For more information about the sample-layout configuration file, see "Sample Layout Configuration File" on page 318.

In layout configuration files, you can use the available tags to specify values for the following elements:

- "Style Configuration Options" on page 304
- "Free-Text Configuration Options" on page 306
- "Image Configuration Options" on page 307
- "Message-Filter Groups" on page 309
- "Health-Gauge Configuration" on page 313

Caution: To see the changes you make to a layout configuration file, exit the web browser and restart the cockpit view. It is not sufficient to only refresh the web browser.

Cockpit views calculate the height of the area reserved for the indicator panel of a cockpit view based on the specifications in the layout configuration files. The Java GUI is added below the indicator panel. It may be hidden from view if the indicator panel takes up all available space. Use the vertical scroll bars of the web browser to access the Java GUI.

Style Configuration Options

The `<styles>` tag enables you to specify global styles for the elements of a cockpit view.

`<styles>`	Colors and font styles for a cockpit view.
`<bg_color>`	Background color of a cockpit view. Example: `<bg_color value="#2e62fe" />`
`<filter_name_font>`	Size and color of the font used for message filters. Example:

	`<filter_name_font size="12" color="#ffffff" />`
`<filter_value_font>`	Size and color of the font used for values in message filters. Example: `<filter_value_font size="11" color="#000000" />`
`<filter_group_font>`	Size and color of the font used for message filter groups. Example: `<filter_group_font size="13" color="#ffffff" />`
`<health_gauge_font>`	Size and color of the font used for health gauges. Example: `<health_gauge_font size="10" color="#ffffff" />`
`<showLabelBackground>`	Whether the background of text areas of message filters and message filter groups displays in color to indicate status. Possible values are true or false. If set to true, the color of the font used for message filters and message filter groups automatically changes to the color specified for *<filter_name_font>*. Example: `<showLabelBackground value="false"/>`
`<showUnowned>`	Whether one or two message bars display. Possible values are `true` or `false`. If set to `false`, only one message bar is displayed. This message bar shows the total number of all messages by severity. If set to `true`, two message bars display. The upper bar shows the total number of all messages by severity. The lower bar shows the number of all unowned messages by severity. Example: `showUnowned value="false"/>`
`<showSeverityIcons>`	Whether state and color indicate the status of message filters and message filter groups. Possible values are `true` or `false`. Example: `<showSeverityIcons value="false"/>`

`<state_color>`	Whether state and color indicate the status of message filters and message filter groups. You can define the following attributes: `state` State of the message filter or message filter group. `value` Color indicating the state of the message filter or message filter group. Examples: `<state_color state="critical" value="#fe0000" />` `<state_color state="major" value="#ff9428" />` `<state_color state="minor" value="#ffde53" />` `<state_color state="warning" value="#4ababc" />` `<state_color state="normal" value="#94cf65" />` `<state_color state="unknown" value="#79a7e2" />` `<state_color state="unowned_critical" value="#fe0000" />` `<state_color state="unowned_major" value="#ff9428" />` `<state_color state="unowned_minor" value="#ffde53" />` `<state_color state="unowned_warning" value="#4ababc" />` `<state_color state="unowned_normal" value="#94cf65" />` `<state_color state="unowned_unknown" value="#79a7e2" />` `<state_color state="no_unowned_messages" value="#b3b3b3" />` `<state_color state="no_owned_messages" value="#dddddd" />` `<state_color state="no_messages" value="#eeeeee" />`

Free-Text Configuration Options

The `<freeTexts>` tag enables you to place single lines of text anywhere in a cockpit view. You can define the position of the line of text, the text itself, and the format of the text.

Note: All styles and attributes not marked *Optional* are required.

`<freeTexts>`	*Optional*. Defines lines of text.
`<text>`	Single line of text. You can define the following attributes: `x` Position of the text line on the x-axis, in pixels, for example: `x="10"`

	`y` Position of the text line on the y-axis, in pixels: `y="10"` `tooltip` *Optional*. Tool tip to display additional information about the line of text, for example: `tooltip="More information."` To access the tool tip of a line of text, hover the cursor over the text. If you do not specify a tool tip, or if the attribute is empty, a tool tip is not available.
`<font>`	*Optional*. Size and color of the font used for the text. You can define the following attributes: `size` *Optional*. Size of the font used for the line of text. For example: `size="10"` `color` *Optional*. Color of the font used for the line of text. For example: `color="#ff0000"` If you do not specify the size and color of the font, the following defaults will be used: `size="10" and color="#FFFFFF"`.
`<b>`	*Optional*. Bold format, for example: `<b>This text appears bold.</b>`
`<u>`	*Optional*. Underline format, for example: `<u>This text appears underlined.</u>`
`<i>`	*Optional*. Italic format, for example: `<i>This text appears in italics.</i>`

Note: You cannot limit the width of a text area (for example, by using line breaks). If a line exceeds the available display area, the cockpit view adds a horizontal scroll bar to the indicator panel of the cockpit view.

Image Configuration Options

The `<images>` tag enables you to place images anywhere in a cockpit view.

Note: All styles and attributes not marked *Optional* are required.

`<images>`	*Optional*. Defines images.
`<image>`	Image. Defines the following attributes:

	`source`	Name of the image file. For more information about the location of the image file, see "Image Location" on page 308. Supported image formats are GIF, JPEG, PNG, SVG, and SWF. Examples: `source="../ITO_OP/images/hp.jpg"` `source="http://mymanager.com/hp.jpg"`
	`width`	Width of the image in pixels, for example: `width="200"`
	`height`	Height of the image in pixels, for example: `height="200"`
	`x`	Position of the image on the x-axis in pixels, for example: `x="10"`
	`y`	Position of the image on the y-axis, in pixels, for example: `y="10"`

Image Location

Depending on the location of the image, you can specify the *absolute* or *relative* path to an image on the management server, or use the HTTP protocol to access images:

- Image location on the management server

  If the image resides in the `/opt/OV/www/htdocs/ito_op/images` directory on the management server, specify the following path in the layout configuration file:

  `../ITO_OP/images/<image>`

  Example use in a layout configuration file:

  `source="../ITO_OP/images/hp.jpg"`

  If the image resides in another location on the management server, specify the absolute path or the relative path, starting from the layout configuration file.

- Image location on any server

  If the image resides on a web server other than the management server, perform the following steps:

  a. Create a cross-domain policy file in the following directory on the management server:

     `/opt/OV/www/htdocs/ito_op/crossdomain.xml`

  b. Add the following lines to the `crossdomain.xml` file:

```
<?xml version="1.0"?>
<!DOCTYPE cross-domain-policy SYSTEM
"http://www.macromedia.com/xml/dtds/cross-domain-policy.dtd">
<cross-domain-policy>
   <allow-access-from domain="<domain>" />
</cross-domain-policy>
```

Replace *<domain>* with the server that hosts the images you want to access, for example `www.mymanager.com`.

Example use in a layout configuration file:

```
source="http://www.mymanager.com/hp.jpg"
```

Message-Filter Groups

The `<messageFilterGroups>` tag enables you to collect message filters into groups and to specify their name, label, position, and other attributes.

Note: All styles and attributes not marked *Optional* are required.

`<messageFilterGroups>`	*Optional*. Defines message-filter groups and message filters.
`<group>`	Name of the message-filter group. You can define the following attributes: `name` — Name of the message-filter group. The name must be unique in the HPOM environment. Note that the group name is not the same as the message-filter name. Filter groups cannot be specified in the Java GUI. Example: `name="corp_srvs"` `label` — *Optional*. Label for the message-filter group. If you do not specify a label, the group `name` is displayed. Example: `label="Corporate Servers"` `tooltip` — *Optional*. Tool tip to display additional information about the message-filter group. To access the tool tip of a message-filter group, hover the cursor over the label or the group summary. If you do not specify any text for the tool tip or the `tooltip` attribute is not

		defined, the string defined in the `label` attribute is displayed in the tool-tip box. If no label is specified, a tool tip is not available. Example: `tooltip="Corporate servers are servicing the company."`
	`textAlign`	Alignment of the name (or label) of the message filter group relative to the message bars. Possible values are `top`, `bottom`, `left`, or `right`. Example: `textAlign="top"`
	`text_width`	Width of the area reserved for displaying the name (or label) of message filters, in pixels. Example: `text_width="130"`
	`bar_width`	Width of the area reserved for displaying the number of messages by severity, in pixels. Choose a suitable width for message bars, depending on the amount of available space and the number of messages expected. A cockpit view requires sufficient space to display message numbers in a human-readable format. If the specified width is too small, the message bars are shaded to indicate that some information is hidden. You can view the missing information by displaying the tool tip provided for a truncated message bar. To hide message bars, specify a negative value (for example, `bar_width="-1"`).

		Example: `bar_width="200"`
	`x`	Position of the message-filter group on the x-axis, in pixels. Example: `x="10"`
	`y`	Position of the message-filter group on the y-axis, in pixels. Example: `y="10"`
	`historyMessages`	Whether to display active or history messages. Possible values are `true` or `false`. Example: `historyMessages="true"`
	`calculateGroupStatus`	Whether to display a summary status line of all messages of the group. Possible values are `true` or `false`. Example: `calculateGroupStatus="true"`
	`showLabelBackground`	*Optional*. Whether to display the background of text areas of message filters and message filter groups in color to indicate status. Possible values are `true` or `false`. If set to true, the color of the font used for message filters and message filter groups automatically changes to the color specified for *<filter_name_font>*. You can also specify `showLabelBackground` globally in the `<styles>` section of a layout

		configuration file. However, any settings that you make at group level override global settings. Example: `showLabelBackground="false"`
	`showSeverityIcons`	*Optional*. Whether to display icons for message filters and message filter groups. Possible values are `true` or `false`. You can also specify `showSeverityIcons` globally in the *`<styles>`* section of a layout configuration file. However, any settings that you make at group level override global settings. Example: `showSeverityIcons="false"`
	`showUnowned`	*Optional*. Whether one or two message bars display. Possible values are `true` or `false`. If set to `false`, only one message bar is displayed. This message bar shows the total number of all messages by severity. If set to `true`, two message bars display. The upper bar shows the total number of all messages by severity. The lower bar shows the number of all unowned messages by severity. **Tip:** You can also specify `showUnowned` globally in the *`<styles>`* section of a layout configuration file. However, any settings that you make at group

	level override global settings. Example: `showUnowned="false"`
`<filter>`	Message filters. You can define the following attributes: `name` — Name of the message filter. The name must correspond to the name of the filter specified in the Java GUI. For details, see the *HPOM Java GUI Operator's Guide*. Example: `name="corp_srv"` `label` — Label for the message filter, for display in the graphical user interface. Example: `label="Corporate Server"` `tooltip` — *Optional*. Tool tip which can be used to display additional information about a message filter. To access the tool tip for a message filter, hover the cursor over the message-filter label in the GUI. If you do not specify any text for the tool tip, or the `tooltip` attribute is not defined, the string defined in the `label` attribute is displayed in the tool-tip box. If no label is specified, a tool tip is not available. Example: `tooltip="This corporate server is servicing the company."`

Health-Gauge Configuration

The *<healthGauges>* tag enables you to define the appearance of the health gauge including: size, position, levels, and scales. For more information about setting the scales of health gauges, see "Defining the Scale of Health Gauges" on page 316.

Note: All styles and attributes not marked *Optional* are required.

`<healthGauges>`	*Optional*. Defines health gauges.
`<gauge>`	Health gauge. You can define the following attributes: `name` — Name of the message filter. The name must correspond to the name of the filter as specified in the Java GUI. For details, see *the HPOM Java GUI Operator's Guide.* Example: `name="corp_srv"` `label` — Label of the health gauge. Example: `label="Corporate Health"` `tooltip` — *Optional*. Tool tip to display additional information about the health gauge. To access the tool tip of a health gauge, hover the cursor over the label. If you do not specify text for the tool tip, or if the attribute is empty, the label displays. If no label is specified, a tool tip is not available. Example: `tooltip="Health gauges show the health of the company."` `sense` — Orientation of the scale. Possible values are `positive` or `negative`. Example: `sense="positive"` `reference` — Maximum value of the scale. Example: `reference="100"`

	`level_1`	Maximum value of the *first* segment on the scale. Example: `level_1="20"`
	`level_2`	Maximum value of the *second* segment on the scale. Example: `level_2="40"`
	`level_3`	Maximum value of the *third* segment on the scale. Example: `level_3="60"`
	`level_4`	Maximum value of the *fourth* segment on the scale. Example: `level_4="80"`
	`width`	Diameter of the health gauge, in pixels. Example: `width="150"`
	`x`	Position of the health gauge on the x-axis, in pixels. Example: `x="20"`
	`y`	Position of the health gauge on the y-axis, in pixels. Example: `y="405"`
	`historyMessages`	Whether to display active or history messages. Possible values are `true` or `false`. Example:

		historyMessages="true"
	showLabelBackground	*Optional*. Whether to display the background of text areas of health gauges in color to indicate status. Possible values are `true` or `false`. If set to true, the color of the font used for health gauges automatically changes to the color specified for *<filter_name_font>*. **Tip:** You can also specify `showLabelBackground` globally in the *<styles>* section of a layout configuration file. However, any settings that you make at gauge level override global settings. Example: `showLabelBackground="false"`
	showSeverityIcons	*Optional*. Whether to display icons for health gauges. Possible values are `true` or `false`. **Tip:** You can also specify `showSeverityIcons` globally in the *<styles>* section of a layout configuration file. However, any settings that you make at gauge level override global settings. Example: `showSeverityIcons"false"`

Defining the Scale of Health Gauges

To define the scale of a health gauge, you must decide the following:

1. General orientation of the scale (positive or negative)
2. Maximum value of the scale
3. Thresholds that define the individual segments of the scale

The `sense` attribute determines the orientation of the scale. A positive orientation means that a lower value is more critical than a higher value, with the reference value being the maximum, the best value. If you choose a negative orientation, higher values are considered to be more critical than lower values.

The `level` attributes define the individual segments of the scale.

Figure 9 shows health gauges with a *positive* orientation:

```
sense="positive"
reference="100" (normal: 80 through 100)
level_1="20" (critical: 0 through 19)
level_2="40" (major: 20 through 39)
level_3="60" (minor: 40 through 59)
level_4="80" (warning: 60 through 79)
```

Figure 9: Health Gauges with a Positive Orientation

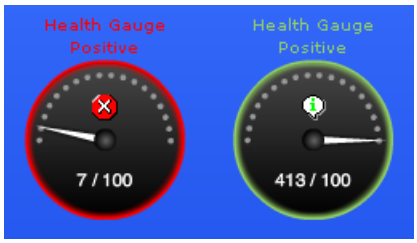


Figure 9 shows health gauges with a *positive* orientation; in a positive orientation, 0=bad, 100=good. The current value of the gauge on the left is `7`, which means that the current status is critical. When the value reaches or exceeds `100`, as shown in the gauge on the right, the status changes to normal because the best possible condition has been reached.

Figure 10 shows health gauges with a *negative* orientation:

```
sense="negative"
reference="100" (critical: 80 through 100)
level_1="20" (normal: 0 through 19)
level_2="40" (warning: 20 through 39)
level_3="60" (minor: 40 through 59)
level_4="80" (major: 60 through 79)
```

Figure 10: Health Gauges with a Negative Orientation

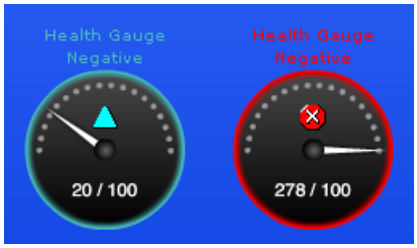


Figure 10 shows health gauges with a *negative* orientation; in a negative orientation, 0=good, 100=bad. The current value of the gauge on the left is 20, which indicates a current status of “warning”. When the value reaches or exceeds 100, as shown in the gauge on the right, the status changes to critical because the worst possible condition has been reached.

Note: If the current value of a health gauge exceeds the reference value, the status of the gauge

remains the same as it was when it reached the reference value. The reference value is: “normal” for health gauges with a positive orientation and “critical” for health gauges with a negative orientation.

Valid Layout Configuration Files

On the management server, the Document Type Definition (DTD) for the layout configuration files is available at the following location:

```
/opt/OV/www/htdocs/ito_op/assets/xml/cockpitviewLayout.dtd
```

HP recommends that you validate your layout configuration files against the DTD provided for this purpose. You must make sure that layout configuration files are well formed XML documents and that conform to the rules of the DTD.

To ensure that the DTD validation tool can locate the cockpit-view DTD (`cockpitviewLayout.dtd`.), insert a reference to the DTD in your XML files, for example:

```
<!DOCTYPE cockpitLayout SYSTEM "/opt/OV/www/htdocs/ito_
op/assets/xml/cockpitviewLayout.dtd">
```

You can find validation tools at the following locations:

- XML Pad:

  http://www.wmhelp.com/
- Eclipse Ganymede (Eclipse IDE for Java Developers):

  http://www.eclipse.org/downloads/packages/release/ganymede/r
- Validome:

  http://www.validome.org/xml/validate/
- W3Schools:

  http://www.w3schools.com

Sample Layout Configuration File

The following sample layout configuration file is available on the management server:

```
/opt/OV/www/htdocs/ito_op/assets/xml/layout_simple.xml
```

The following two message filters are installed by default in the Java GUI:

Caution: These filters are required to successfully view the sample layout configuration file. If you want to keep the default layout, do not edit or remove them.

- Filter `Main Terminal` (shows all messages)
- Filter `WebShop DB` (shows messages with severity minor, major or critical)

Tip: Do *not* edit the sample layout-configuration file directly. First make a copy of the file, and then edit the new copy.

```
<?xml version="1.0" encoding="utf-8" ?>
<!DOCTYPE cockpitLayout SYSTEM "/opt/OV/www/htdocs/ito_op/
assets/xml/cockpitviewLayout.dtd">

<cockpitLayout>
<layoutVersion version="001.001" />

<styles>
   <bg_color value="#2e62fe" />
   <filter_name_font size="12" color="#ffffff" />
   <filter_value_font size="11" color="#000000" />
   <filter_group_font size="13" color="#ffffff" />
   <health_gauge_font size="10" color="#ffffff" />
   <showLabelBackground value="false" />
   <showSeverityIcons value="true" />
   <showUnowned value="true" />
   <state_color state="critical" value="#fe0000" />
   <state_color state="major" value="#ff9428" />
   <state_color state="minor" value="#ffde53" />
   <state_color state="warning" value="#4ababc" />
   <state_color state="normal" value="#94cf65" />
   <state_color state="unknown" value="#79a7e2" />
   <state_color state="unowned_critical" value="#fe0000" />
   <state_color state="unowned_major" value="#ff9428" />
   <state_color state="unowned_minor" value="#ffde53" />
   <state_color state="unowned_warning" value="#4ababc" />
   <state_color state="unowned_normal" value="#94cf65" />
   <state_color state="unowned_unknown" value="#79a7e2" />
   <state_color state="no_unowned_messages" value="#b3b3b3" />
   <state_color state="no_owned_messages" value="#dddddd" />
   <state_color state="no_messages" value="#b3b3b3" />
</styles>

<freeTexts>
   <text x="110" y="380">
      <u>
         <b>Lorem ipsum</b>
      </u>
      dolor sit amet,
      <u>consectetuer</u>
      <font size="14" color="#ff0000">adipiscing</font>
      elit
   </text>
```

```
</freeTexts>

<messageFilterGroups>
   <group name="Corporate Servers" label="Corporate Servers"
   text_width="120" bar_width="200" x="10" y="10"
   calculateGroupStatus="true">
      <filter name="WebShop DB" label="WebShop DB" />
      <filter name="Main Terminal" label="Main Terminal" />
   </group>
</messageFilterGroups>

<healthGauges>
   <gauge name="Main Terminal" label="Main Terminal"
   sense="positive" reference="12" level_1="60" level_2="70"
   level_3="80" level_4="100" width="110" x="20" y="205" />
   <gauge name="WebShop DB" label="WebShop DB" sense="negative"
   reference="10" level_1="30" level_2="35" level_3="40"
   level_4="44" width="110" x="210" y="205" />
</healthGauges>

<images>
   <image source="../ITO_OP/images/hp.jpg" width="100"
   height="60" x="0" y="350" />
</images>

</cockpitLayout>
```

Backup Management Servers

If the currently connected HP Operations management server suddenly becomes unavailable, for example because of a system failure, Java GUI clients can automatically reconnect to one or more backup management servers.

If the connection is disrupted, the Java GUI tries to connect to the current HP Operations management server by default three times. If all reconnects fail, Java GUI users are asked whether they want to connect to the next backup management server in the list or continue trying to connect to the current management server. If they choose the current management server, the Java GUI will try to connect until the server can be reached again or until the Java GUI is closed.

If the user names and passwords of the connecting HPOM users are known on all participating management servers, the Java GUI reconnects to a backup server without displaying the `Login` dialog box.

You can configure the number and order of backup management servers for each HP Operations management server, as well as the number of reconnect attempts of the Java GUI client by setting parameters for the `ovconfchg` command line tool:

- Backup management servers:

  The keyword OPC_JGUI_BACKUP_SRV enables you to create a list of HPOM backup management servers that provide connections for Java GUIs. Use commas or colons to separate the management server host names.

  In the following example, the HP Operations management servers ovo1.hp.com and ovo2.hp.com are configured as backup servers for all connecting Java GUIs:

  ```
  ovconfchg -ovrg server -ns opc -set OPC_JGUI_BACKUP_SRV ovo1.hp.com,ovo2.hp.com
  ```

- Number of reconnect attempts:

  The keyword OPC_JGUI_RECONNECT_RETRIES specifies the number of times a Java GUI client attempts to connect to the primary HPOM management server before trying to connect to a backup management server.

  In the following example, the maximum number of reconnect attempts is configured to be five.

  ```
  ovconfchg -ovrg server -ns opc -set OPC_JGUI_RECONNECT_RETRIES 5
  ```

The Java GUI must be restarted after the configuration has been updated on the management server. For more information about command options and parameters, see the *ovconfchg(1)* manual page.

Java GUI APIs

HPOM enables you to control certain Java GUI features remotely from other Java applications using the Java GUI Remote application programming interface (API).

For more information about the remote APIs that are available for the Java GUI, see the Java GUI Remote APIs Specification, which you can find on the HPOM management server at the following location:

- Standard connection:

  ```
  http://<management_server>:8081/ITO_DOC
  ```

- Secure connection:

  ```
  https://<management_server>:8444/ITO_DOC
  ```

In this instance, *<management_server>* is the fully qualified hostname of your HPOM management server.

Enabling Java GUI Remote APIs

To enable Java GUI remote APIs, follow these steps:

1. On the HP Operations management server, set the JGUI_API_ENABLED server configuration variable to TRUE by running the following command:

```
/opt/OV/bin/ovconfchg -ovrg server -ns opc -set JGUI_API_ENABLED TRUE
```

2. Restart Service Navigator.

Global Property Files

HPOM stores custom changes to the Java GUI in a selection of property files, which reside in the home directory of the operating-system user who launches the Java GUI. The property files include the following files:

- Console settings:
  - `HP_OV_consoleSettings_<server_name>_<user>`
  - `HP_OV_consoleSettings_<server_name>`
  - `HP_OV_consoleSettings`

  For more information about saving console settings, see the *HPOM Java GUI Operator's Guide*.
- Resources:

  The Java GUI resource file `itooprc`. For more information about the resource file for the Java GUI, see "Resource Files" on page 295.
- Browser settings:

  The browser settings are stored in the file `itoopbrw`. For more information about the location and contents of the browser settings file for the Java GUI, see the *HPOM Java GUI Operator's Guide*.

You can configure the Java GUI to override any individual settings and use global property files stored in a shared location. The settings configured in a global property files override any individual settings with the following exceptions:

- Startup parameters:

  The following parameters control the connection to the HPOM management server and are ignored in global mode:
  - `initial_node`
  - `user`
  - `passwd`
  - `port`
  - `locale`
- Allowed users:

The Java GUI continues to use the individual property files of the administrator or operators, if such files exist in the user's home directory. See also "Individual Settings with Global Property Files" on page 323.

Enabling Global Property Files

To enable global property files for the Java GUI, use the `ovconfchg` configuration tool on the HP Operations management server as follows:

1. Create a shared location where the global property files are stored.

   The shared location can be one of the following:

   - Local path:

     Examples: `X:\share\javagui` or `/net/share/javagui`

   - Remote path:

     Example: `\\jacko.hp.com\share\javagui`

   - URL:

     Must start with the string "`http:`" or "`https:`", for example: `http://jacko:8081/ITO_OP/` or `https://jacko:8444/ITO_OP/`

2. Copy the global property files to the shared location.
3. Configure the Java GUI to evaluate the global property files on the host operating system:

   - Windows:

     ```
     ovconfchg -ovrg server -ns opc -set OPC_JGUI_GLOBAL_SETTINGS_WIN <win_shared_location>
     ```

   - UNIX:

     ```
     ovconfchg -ovrg server -ns opc -set OPC_JGUI_GLOBAL_SETTINGS_UNIX <unix_shared_location>
     ```

   The Java GUI clients running on Windows systems will read the global settings from the location specified in the `OPC_JGUI_GLOBAL_SETTINGS_WIN` variable, while clients running on UNIX systems will read the global settings from the location specified in the `OPC_JGUI_GLOBAL_SETTINGS_UNIX` variable.
4. Restart all running Java GUI clients.

Individual Settings with Global Property Files

When global property files are enabled and configured, only the administrator and, if so configured, selected operators, are allowed to save and use individual settings. These users can save their

settings in their home directories without affecting the global settings files.

Authorizing Access to Property Files

To grant permission to selected operators to save and use individual property files, specify their user names, separated by commas, as options for the variable `OPC_JGUI_CONF_ALLOWED_USERS`, as follows:

```
# /opt/OV/bin/ovconfchg -ovrg server -ns opc -set OPC_JGUI_CONF_ALLOWED_USERS opc_
op,itoop
```

For all users that are treated as allowed users, the property files in their local home directories are evaluated first, if they exist. Then the global property files are loaded from the shared location.

Global Configuration Change Notifications

By default, Java GUI clients check every five minutes for changes to the global property files in the shared location. If a change is detected, the `OVO Communication Status` dialog box displays a message, which informs the operator of the changes and requests a restart of the Java GUI.

You can change the polling interval by specifying a value for the parameter `global_settings_poll_interval` in the `itooprc` file.

Setting the Change-Notification Polling Interval

To set the change-notification polling interval to one minute, add the following line to the `itooprc` file:

```
global_settings_poll_interval 1
```

Secure HTTPS-based Communication

The HTTPS-based Java GUI uses a combination of secure HTTP (HTTPS) and Secure Socket Layer (SSL) encryption for all communication with the HP Operations management server. The SSL encryption is based on the Core functionality components.

For more information about the architecture of the HTTPS-based Java GUI as well as configuration and usage, see the *HPOM Java GUI Operator's Guide*.

For more information about installing and enabling the HTTPS based Java GUI as well as disabling standard (unsecured) communication between the Java GUI client and the HP Operations management server, see the *HPOM Installation Guide for the Management Server*.

Secure Communication Setup

The process of establishing a secure connection between the Java GUI client and the `opcuihttps` process on the HP Operations management server, is as follows:

1. The Java GUI client connects to the `opcuihttps` process, which acts as a proxy between Java GUI client and HP Operations management server using the HTTPS protocol.
2. The Java GUI communicates with `opcuihttps` process using the secure HTTP protocol (HTTPS) on port 35211. The `opcuihttps` then redirects the HTTPS requests to the standard Java GUI port (2531) using socket communication.

   If your firewall is configured to allow outbound connections only or you do not want to open additional ports (such as 35211) on the firewall, you can configure a secure connection between the Java GUI and `opcuihttps` using the port already in use by the communication broker `ovbbccb` (port 383) provided certain prerequisites are met on the management server and the host running the Java GUI client. For more information, see "Secure Outbound Connections with ovbbccb" on page 326.

   Note: Make sure the port to which the HTTPS requests are redirected is set to the default value 2531. The option for connecting the `opcuihttps` process to other than default `opcuiwww` port is currently *not* available.
3. All forwarded HTTPS requests are then handled by the `inetd` process (on HP-UX and Solaris) or `xinetd` process (on Linux), as well as the requests from non-secure Java GUI clients.
4. The `opcuihttps` also processes replies from the HP Operations management server and forwards them to the Java GUI using the HTTPS protocol. All communication requests in either direction between the Java GUI and the HPOM management server become trustworthy for secure exchange of data.

For more information about how to configure `opcuihttps` settings and display a list of the parameters related to HTTPS based Java GUI, see "opcuihttps Configuration" on page 327.

Figure 11 shows the communication between client and server. Depending on the chosen communication type, the following applies:

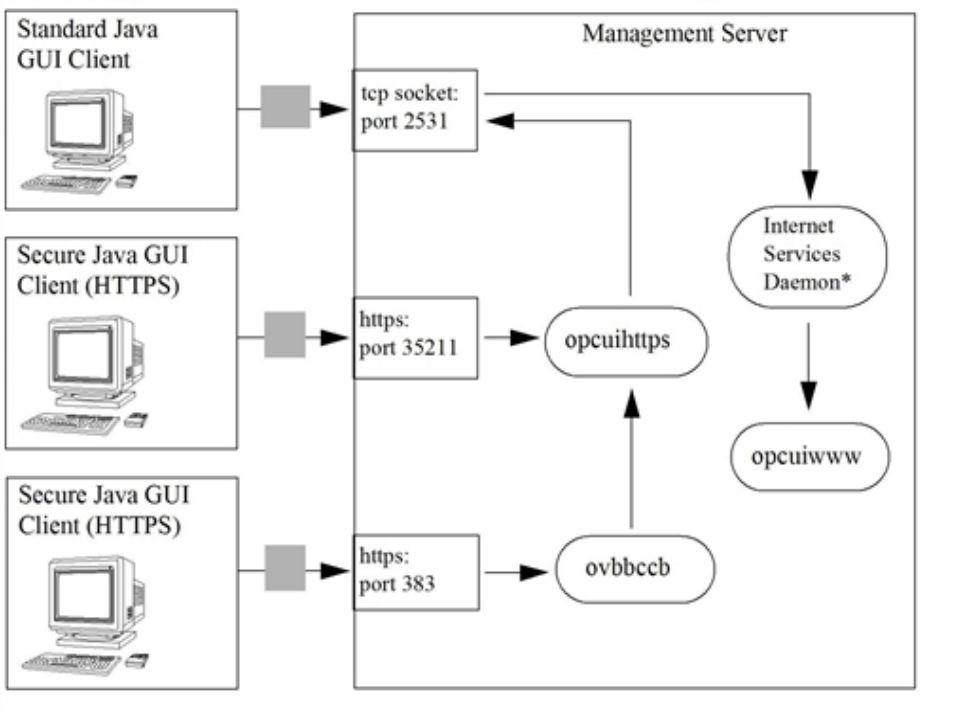
- HTTPS-based communication:

  If you are using secure, HTTPS-based communication, a *locked* padlock icon appears in the logon window for the Java GUI client and in the status bar of the running Java GUI client.
- Standard communication:

  If you are using standard (insecure) HTTP-based communication, an *open* padlock icon appears in status bar of the running Java GUI client.

In Figure 11, references to the Internet Services Daemon (*) mean `inetd` on UNIX and `xinetd` on Linux.

Figure 11: Connection Between Java GUI Client and HPOM Server



For more information about the authentication process that ensures the establishment of a secure connection between the Java GUI client and the HPOM management server, including the provision and installation of certificates, see the *HPOM Java GUI Operator's Guide*.

Secure Outbound Connections with ovbbccb

If your firewall is configured to allow outbound connections only or you do not want to open additional ports (such as 35211), you can configure the HTTPS-based Java GUI to establish a secure connection with `opcuihttps` using the communication broker `ovbbccb` on the HPOM management server (through port number 383 which is already open for use by `ovbbccb`). However, the following prerequisites apply:

- You must set the secure-connection mode for the `opcuihttps` process to `RequireCertificate`. On the HPOM management server, use the `ovconfchg` command to set an XPL variable as follows:

  ```
  # ovconfchg -ovrg server -ns opc.opcuihttps -set SSL_CLIENT_VERIFICATION_MODE
  RequireCertificate
  ```

  For more information about the parameters available to configure `opcuihttps`, see "ovconfchg Parameters for the opcuihttps Name Space" on page 328.

- On the system where the Java GUI client is running, install one of the following components:
  - An HPOM client certificate:

    A certificate is required for outbound connections in both anonymous and full-authentication verification modes.

  - An HPOM HTTPS agent.
- On the system where the Java GUI client is running, add the following settings to either the `itooprc` file or the `ito_op.bat` file, which are used to specify a Java GUI profile:

  ```
  https true
  lcore_defaults true
  https_port 383
  ```

  If `lcore_defaults` is set to `true` in the Java GUI profile, the Java GUI looks for and uses the HPOM agent's security certificate for authentication. However, if no agent is installed, the Java GUI can use a manually generated certificate stored in the same (agent-default) location. For more information about the parameters you can use in the `itooprc` resource file, see "itooprc Options and Parameters" on page 295.

opcuihttps Configuration

The `opcuihttps` process acts as a proxy between the Java GUI client and the HP Operations management server. Controlled by the HPOM server Control process, `ovcd`, `opcuihttps` is started and stopped together with the other server processes.

The `opcuihttps` binary is installed in the `/opt/OV/bin/OpC` directory. The configuration parameters for `opcuihttps` are read at startup. For more information about using the `ovconfchg` command to change the run-time parameters for `opcuihttps`, see "Changing HTTPS Parameters" on page 327 or the *ovconfchg(1)* manual page.

Changing HTTPS Parameters

To change the `opcuihttps` parameters that define the communication between HPOM management server and the Java GUI client, perform the following steps:

1. Use the `ovconfchg` command to set a parameter for the `opcuihttps` name space, as follows:

   ```
   # ovconfchg -ovrg server -ns opc.opcuihttps -set <parameter> <value>
   ```

   For more information about the `ovconfchg` command, see the *ovconfchg(1)* manual page. Table 20 lists the parameters you can use to configure the `opcuihttps` process.
2. If any of the `opcuihttps` parameters are changed at runtime, you must restart the `opcuihttps` process.

Table 20 lists the parameters that you can use to configure the `opcuihttps` process.

Table 20: ovconfchg Parameters for the opcuihttps Name Space

Parameter	Format	Default value	Description
SERVER_PORT [a]	*<number>*	35211[b]	Port on which the Java GUI is listening.
OPCUIWWW_PORT	*<number>*	2531	opcuiwww port number as defined in the entry ito-e-gui in the /etc/services file.
SSL_CLIENT_ VERIFICATION_MODE	Anonymous| RequireCertificate	Anonymous	Whether the opcuihttps server accepts anonymous connections from the Java GUI clients (or the communication broker ovbbccb). If set to RequireCertificate, the clients require a certificate that permits full authentication [c].
MAX_CONNECTIONS	*<number>*	100	Maximum number of connections allowed to opcuihttps.

Note: You can check if it is possible to connect to the opcuihttps process using a Web browser, such as Microsoft Internet Explorer or Mozilla by entering the following URL in your browser:

https://*<server>*:*<port>*/opcuihttps/info

In the example URL, <server> is an HP Operations management server hostname and <port> is the port on which opcuihttps is listening.

Secure Java GUI Connections

For the HTTPS based Java GUI to communicate with an HPOM management server through a firewall, you can configure either of the following components:

[a]For troubleshooting purposes, you can also set the port on the command line by starting opcuihttps with the specified *<server_port>* parameter.

[b]The port on which opcuihttps is listening for requests to establish a secure HTTPS-based connection. The standard (insecure) Java GUI uses port 2531.

[c]Set lcore_defaults true if you want the Java GUI to use the HPOM agent's security certificate for authentication or a manually generated certificate stored in the same (agent-default) location.

- Firewall:

  Allow the HTTPS based Java GUI direct access to the HPOM management server. Note that if you connect the Java GUI to the HPOM management server through the communication broker `ovbbccb` on port number 383, you do not need to open any additional ports.

  For more information about using the `ovbbccb` to connect the Java GUI to the `opcuihttps` process on the HPOM management server, see "Secure Outbound Connections with ovbbccb" on page 326.

- HTTPS based Java GUI:

  The HTTPS-based Java GUI uses a proxy server for all communication with the HPOM management server. Figure 11 illustrates the default port (35211) on which the `opcuihttps` process listens for requests from a Java GUI client to establish a *secure* connection with the management server. [a]

  Secure connections between the Java GUI client and the HPOM management server are also possible through the communication broker `ovbbccb` on port number 383 provided certain configuration prerequisites are met on both the HPOM management server and the host where the Java GUI client is running. For more information about using the `ovbbccb` to connect the Java GUI to the `opcuihttps` process on the HPOM management server, see "Secure Outbound Connections with ovbbccb" on page 326.

  There are several different methods for specifying a proxy server for the HTTPS based Java GUI:

  - `ito_op` command-line tool
  - `itooprc` file
  - `Login` dialog box
  - Java GUI applets
  - Core functionality

For more information about the various methods for specifying Java-GUI connections through a proxy server, see the *HPOM Java GUI Operator's Guide*.

Preventing HTTPS Timeouts by Configuring Message Loading in Multiple Chunks

With HTTPS-based communication, some specific timeouts are defined for HTTPS requests and replies. These timeouts can also occur as a result of huge replies, such as when a huge number of messages is sent from `opcuiwww` to the Java GUI message in a single chunk.

[a]Note that the Java GUI uses port 3521 for standard (insecure) connections.

However, it is possible to avoid these timeouts from occurring by using the new mechanism that transfers messages in multiple chunks, which contain a predefined number of messages. So, instead of one having one huge reply, multiple requests and replies are executed until all required messages are transferred.

Note: This feature does *not* guarantee that https timeouts will not happen. They can still happen due to long-lasting database operations or other expected factors (network problems). However, loading of messages in multiple chunks shortens the time until the first reply is sent back to the Java GUI, which eliminates lot of unnecessary timeouts.

Configuring Message Loading in Multiple Chunks

To configure message loading in multiple chunks, set the following server parameters as appropriate:

- To enable the feature, type the following:

  ```
  ovconfchg -ovrg server -ns opc -set OPCUIWWW_MSG_CHUNK_MODE
  ```

  Where the possible values are `TRUE` or `FALSE`. The default value is `FALSE`.
- To set the number of messages to be loaded in one chunk, type the following:

  ```
  ovconfchg -ovrg server -ns opc -set OPCUIWWW_MSG_CHUNK_SIZE
  ```

  Where the possible values are from 50 through `maxint` (232 -1).
  The default value is 1000 messages.

Defining a Tool Timeout

To define a tool timeout, run the following commands:

```
/opt/OV/bin/ovconfchg -ovrg server -ns opc -set OPCUIWWW_KILL_APP TRUE
```

```
/opt/OV/bin/ovconfchg -ovrg server -ns opc -set OPCUIWWW_KILL_APP_TIMEOUT <number_of_minutes>
```

In this instance, *<number_of_minutes>* is the number of minutes after which an unresponsive tool is stopped. The default value is 10.

For example, to set the tool timeout to one minute, run the following commands:

```
/opt/OV/bin/ovconfchg -ovrg server -ns opc -set OPCUIWWW_KILL_APP TRUE
```

```
/opt/OV/bin/ovconfchg -ovrg server -ns opc -set OPCUIWWW_KILL_APP_TIMEOUT 1
```

If the tool did not finish running and the timeout expires, the tool is stopped and the following error message appears:

```
Tool halted due to timeout - resume is not possible.
This window can be closed.
```

Caution: The default value of `OPCUIWWW_KILL_APP` is `FALSE`. Therefore, make sure that you set this server configuration variable to `TRUE` before you define the number of minutes after which the tool is stopped.

Setting a Size Limit for the opcuiwww.sh.log File

To set a size limit for the `/var/opt/OV/log/opcuiwww.sh.log` file, use the `OPCUIWWW_LOG_SIZE` server configuration variable. This variable indicates the maximum log file size in kilobytes. When the `opcuiwww.sh.log` file exceeds the specified size, its contents are moved to the `opcuiwww.sh.log.old` file and the `opcuiwww.sh.log` file is started anew.

For example, to set the maximum log file size to 1 KB, run the following command:

```
/opt/OV/bin/ovconfchg -ovrg server -ns opc -set OPCUIWWW_LOG_SIZE 1
```

Note: The value of the `OPCUIWWW_LOG_SIZE` server configuration variable must be an integer—the minimum value is 1 KB, whereas the maximum value is not limited. If the maximum log file size is not set correctly (for example, it is set to `-1`) or is not set at all, the default value from the `opcuiwww.sh` file (that is, 1024 KB) is used.

Operator Defaults

As an HPOM administrator, you can define default startup behavior for operator areas in Java GUI with two application groups:

- Shortcuts:

  You can create new application groups that are added individually at the end of the Java GUI shortcut bar. These application groups can contain any kind of application.
- Work spaces:

  You can create new application groups that are added individually after existing default workspaces in the Java GUI workspace pane. These application groups can contain any kind of application.

Note: You can assign a set of shortcuts or work spaces to an individual operator, a group of operators, or all operators.

For more information about operator defaults assigned by the HPOM administrator, see the *HPOM Java GUI Operator's Guide*.

Assigning Operator Defaults

To assign operator defaults, you must be familiar with the following procedures:

1. Create application groups using the `opcappl` command-line tool.
2. Add applications to the application groups using the `opcappl` command-line tool.

   If you want to enable users to start applications as local applications in the Java GUI, specify the application call value as follows:

   - Windows:

     `app_call="cmd /c start <application_name>"`

   - Linux:

     `app_call="xterm -e <application_name>"`

   - UNIX:

     `app_call="dtterm -e <application_name>"`

   For example, to enable starting `telnet` on Windows, enter the following command:

   ```
   # opcappl -add_app app_name=APP_X app_call="cmd /c start telnet $OPC_NODES"
   user_name=John passwd=xyz
   ```

3. Assign applications and application groups using the `opccfguser` command.

   Note: When you assign an application with a hierarchical structure, that is an application group, the same structure is assigned to an operator.

For more information about command options and parameters, see the *opcappl(1m)* and *opccfguser (1m)* manual pages.

Allowing or Denying Access to Java GUI Clients

To specify which Java GUI operators or hostnames are allowed or not allowed to connect to the HP Operations management server, use the following configuration variables:

- `OPC_JGUI_ALLOWED_OPERATOR`
- `OPC_JGUI_DENIED_OPERATOR`
- `OPC_JGUI_ALLOWED_HOSTNAME`
- `OPC_JGUI_DENIED_HOSTNAME`

You can also define multiple values. In this case, make sure to separate them by a comma or a colon (whitespace is trimmed).

Examples:

```
ovconfchg -ovrg server -ns opc -set OPC_JGUI_ALLOWED_OPERATOR opc_op,george,anna
ovconfchg -ovrg server -ns opc -set OPC_JGUI_DENIED_HOSTNAME rose.hp.com
```

Note: If an operator or a hostname is not allowed to connect, an error message appears.

When defining a hostname, choose one of the following:

- full hostname (for example, `machine.domain.com`)
- short hostname (for example, `machine`)
- IP address (for example, `10.20.30.40`)
- IP pattern (for example, `10.20.30.*`, `10.20.*.*`, `*.*.*.*`, and so on)

If you choose a full or short hostname, make sure that the hostname is resolvable to the IP address (by the local DNS). Otherwise, a warning message appears. If you choose an IP pattern, make sure that it has the same number of dots as the local IP address. Otherwise, a warning message appears.

Caution: If the `OPC_JGUI_ALLOWED_OPERATOR` configuration variable is defined, the `OPC_JGUI_DENIED_OPERATOR` configuration variable is ignored. If the `OPC_JGUI_ALLOWED_HOSTNAME` configuration variable is defined, the `OPC_JGUI_DENIED_HOSTNAME` configuration variable is ignored.

For detailed information about configuration variables, see the *HPOM Server Configuration Variables* document.

Custom Message-Group Icons

You can customize message-group icons by using the server-side variable `OPC_JGUI_MSGGRP_ICON` in one of the following ways:

- Icon color:

  Display the default message-group icon in monochrome (black and white). For more information about changing the icon color, see "Changing the Icon Color" on page 333.

- Icon image:

  Load a custom image. For more information about changing the icon image source, see "Changing the Icon Image" on page 334.

- Icon severity:

  Load an empty image but retain the severity status (for example, red for critical). For more information about retaining the icon severity, see "Retaining Icon Severity Status" on page 334.

Changing the Icon Color

To change the color of a default icon in the Java GUI from color to monochrome (black and white), use the `ovconfchg` command to set the OPC_JGUI_MSGGRP_ICON, as follows:

```
# ovconfchg -ovrg server -ns opc -set OPC_JGUI_MSGGRP_ICON=BW
```

Changing the Icon Image

To replace the default icon image with a custom image use the OPC_JGUI_MSGGRP_ICON variable to specify the path to the new image file and use the ovconfchg command to set the variable, as follows:

```
# ovconfchg -ovrg server -ns opc -set OPC_JGUI_MSGGRP_ICON=http://<HPOM_
server>:8081/ITO_OP/images/<file_name>.32.gif
```

or

```
# ovconfchg -ovrg server -ns opc -set OPC_JGUI_MSGGRP_ICON=https://<HPOM_
server>:8444/ITO_OP/images/<file_name>.32.gif
```

<HPOM_server>	Name of the HPOM management server where the custom image files are stored.
<file_server>	Name of the custom image file you want to use to replace the default message-group icon.

Retaining Icon Severity Status

To load an empty image but retain the original severity status (for example, red for critical) use the ovconfchg command to set the OPC_JGUI_MSGGRP_ICON variable to "nonexisting_image", as follows:

```
# ovconfchg -ovrg server -ns opc -set OPC_JGUI_MSGGRP_ICON=nonexisting_image
```

Setting Severity Labels

To set the severity labels according to your preferences, you can use the following server configuration variables:

- OPC_JGUI_SEV_EN
- OPC_JGUI_SEV_ES
- OPC_JGUI_SEV_KO
- OPC_JGUI_SEV_CN
- OPC_JGUI_SEV_JA

Each of these server configuration variables corresponds to one of the supported locales and is loaded when the locale is selected. The server configuration variable must have a value that consists of six severity labels separated by commas and listed in the following severity order: unknown, normal, warning, minor, major, and critical.

For example:

```
/opt/OV/bin/ovconfchg -ovrg server -ns opc -set OPC_JGUI_SEV_EN
this,is,new,custom,severity,label
```

In this instance, the severity labels are changed as described in Table 21.

Table 21: Severity Label Change

Default Severity Label	New Severity Label
unknown	this
normal	is
warning	new
minor	custom
major	severity
critical	label

If you do not want to change one or more severity labels, use the ? character instead of a severity label. For example:

```
/opt/OV/bin/ovconfchg -ovrg server -ns opc -set OPC_JGUI_SEV_EN
this,is,?,custom,severity,?
```

In this instance, each occurrence of the ? character is ignored and the default severity label is loaded.

If you want one or more severity labels to contain white spaces, the whole server configuration variable must be within straight quotation marks. For example:

```
/opt/OV/bin/ovconfchg -ovrg server -ns opc -set OPC_JGUI_SEV_EN
"unknown,normal,custom warning,my minor,major,critical"
```

If the ? character is used within the quoted server configuration variable value, the default severity label is loaded for the corresponding severity. For example:

```
/opt/OV/bin/ovconfchg -ovrg server -ns opc -set OPC_JGUI_SEV_EN
"unknown,normal,?,my minor,major,critical"
```

Note: If the value of the server configuration variable has fewer or more than six labels, the value is ignored and the default severity labels are loaded.

If you want to restore the default severity labels, delete the corresponding server configuration variable. For example:

```
/opt/OV/bin/ovconfchg -ovrg server -ns opc -clear OPC_JGUI_SEV_EN
```

Caution: For the changes to take effect, **File->Reload Configuration** must be selected.

Client Version Control

The Java GUI client version control feature enables you to use server configuration variables to specify which versions of the Java GUI are required or recommended. This feature allows you to configure the HP Operations management server to approve or deny connection requests from Java GUI clients on the basis of the Java GUI client version.

The following server configuration variables are available for specifying the Java GUI client version:

OPC_JGUI_MINIMAL_VER	The minimum required version of the Java GUI client that can connect to the HP Operations management server.
OPC_JGUI_RECOMMENDED_VER	The minimum recommended version of the Java GUI client that can connect to the HP Operations management server.

Note: To prevent a hyperlink from appearing in the pop-up dialog when the Java GUI client version control feature used, set the OPC_JGUI_VER_DOWNLOAD_URL configuration variable to NONE.

Specifying the Required Java GUI Client Version

To specify which version of the Java GUI client is permitted to connect to the HP Operations management server, use the ovconfchg command with the following parameters and options:

```
# ovconfchg -ovrg server -ns opc -set OPC_JGUI_MINIMAL_VER 09.10.200
```

In this instance, Java GUI clients with versions lower than 09.10.200 cannot connect to the management server.

Specifying the Recommended Java GUI Client Version

To specify which version of the Java GUI client is recommended for any connection to the HP Operations management server, use the ovconfchg command with the following parameters and options:

```
# ovconfchg -ovrg server -ns opc -set OPC_JGUI_RECOMMENDED_VER 09.10.220
```

In this instance, the recommended Java GUI client version is 09.10.220. However, Java GUI clients with versions lower than the recommended version can still connect to the management server.

You can combine the configurations illustrated in these examples to make sure that the HP Operations management server will only accept connections from the recommended Java GUI client version and the minimum required Java GUI client version.

Note: Connections from Java GUI clients that are allowed but not recommended display a message that informs the operator which version of the Java GUI is recommended for connections to the selected management server. Connection requests from Java GUI clients other than the allowed or recommended versions are refused.

Specifying Exceptions to Permitted Java GUI Client Versions

To prevent Java GUI clients with versions lower than 09.10.200 from connecting to the management server with the exception of 09.10.152.QXCR1000xxxxxx, use the `ovconfchg` command with the following parameters and options:

```
# ovconfchg -ovrg server -ns opc -set OPC_JGUI_MINIMAL_VER
09.10.200,09.10.152.QXCR1000xxxxxx
```

Note: The `listguis` command line interface shows the Java GUI client version.

Tips and Tricks

The information in this section is designed to help you improve the overall performance of the HPOM Java-based operator GUI. The information provided covers the following areas:

- "User Sessions " on page 337
- "Security Exceptions" on page 338
- "Messages and Message IDs" on page 339

User Sessions

Before stopping the HP Operations management server or the database processes for any significant period of time, it is polite and helpful to identify the HPOM operators who are currently logged into HPOM with the Java GUI so that you can notify them of the planned outage.

Listing Java GUI Connections

To find out who is currently logged into HPOM with the Java GUI, use the `listguis` tool with the `-java` parameter as follows:

```
# /opt/OV/contrib/OpC/listguis -java
```

The command output lists the number of currently open Java GUI sessions and the following additional information:

mode	Process used by Java GUI connection, for example: m — Master mode is the main process for the Java GUI for a specific user. c — Channel mode is a subprocess that forwards requests to the master mode (m). ct — Client tool mode is a subprocess that forwards requests to the master mode (m).
PID	Process ID for the open connection, for example: 9110
Operator Name	Name of the user logged in with the Java GUI, for example, opc_adm.
GUI hostname	Hostname of the machine where the Java GUI is running, for example: omlux.hp.com
GUI IP Address	IP address of the machine where the Java GUI is running, for example: 15.16.17.180.
HTTPS	Standard or secure connection (HTTP/S) between the Java GUI and the HPOM management server. For example, Yes (HTTPS) or No (HTTP).
GUI Port	Port on the HPOM management server that the Java GUI is using to connect to HPOM, for example: 35211
GUI Version	Version of the Java GUI client connected to the HPOM management server, for example: 09.01.180.
Since	Time at which the current connection from the Java GUI to the HPOM management server was first established, for example: 07:14.
%CPU	Amount of CPU required to run the current connection, for example: 0.0

You can use this information to contact the operators and ask them to close the Java GUI or, if necessary, kill the opcuiwww processes remotely.

Security Exceptions

If you receive a security exception warning when trying to run the Java GUI as an applet in a web browser, it is highly likely that the security file identitydb.obj is either missing or corrupt, for example, because it was not downloaded in binary mode.

Downloading the identitydb.obj Security File

To download the security file `identitydb.obj` in binary mode, follow these steps.

1. Open the file `/opt/OV/httpd/conf/mime.types`, and add the following line:

   ```
   application/x-javakey        obj
   ```

2. As user root, restart your Apache web server using the following command:

   ```
   # /opt/OV/httpd/bin/apachectl restart
   ```

3. Download the file `identitydb.obj` again.

Messages and Message IDs

By default, the Java GUI uses the complete contents of new (active) messages for all internal communication and processing instead of just the message ID. Using the complete contents of a message improves overall system performance by reducing the number and frequency of read-write requests to the database.

Configuring the Internal Use of Complete Messages

To ensure that you HPOM always uses the complete contents of new messages for internal communication rather than only the message ID, check that the configuration variable for `opcuiwww` is either not set or explicitly set to `TRUE`, as follows:

1. Check the current configuration settings using the `ovconfget` command, as follows:

   ```
   # ovconfget -ovrg server -ns opc
   ```

   The `ovconfget` command displays the content included in the `opc` section of the `local_settings.ini` file, for example:

   ```
   [opc]
   DATABASE=ov_net
   OPCUIWWW_NEW_MSG_NO_DB=FALSE
   OPC_HA=FALSE
   OPC_INSTALLATION_TIME=09/04/09 13:12:18
   OPC_INSTALLED_VERSION=09.01.180
   OPC_MGMTSV_CHARSET=utf8
   OPC_MGMT_SERVER=omlux1.hp.com
   OPC_SVCM_ADD_WARN_IF_EXISTS=TRUE
   OPC_SVCM_ERROR_CHECKING=FULL
   ```

2. Make sure that the variable `OPCUIWWW_NEW_MSG_NO_DB` is *not* set to `FALSE`.
3. To set the variable `OPCUIWWW_NEW_MSG_NO_DB` explicitly to `TRUE` in the `opc` name space, use the `ovconfchg` command on the HPOM management server with the `-set` parameter, as follows:

```
# ovconfchg -ovrg server -ns opc -set OPCUIWWW_NEW_MSG_NO_DB TRUE
```

4. Check the new configuration settings using the `ovconfget` command, as follows:

   ```
   # ovconfget -ovrg server opc
   ```

   The command displays the updated content from the `opc` section of the `local_settings.ini` file including the new setting for the `OPCUIWWW_NEW_MSG_NO_DB` variable, for example:

   ```
   [opc]
   DATABASE=ov_net
   OPCUIWWW_NEW_MSG_NO_DB=TRUE
   ...
   ```

For more information about the `ovconfget` and `ovconfchg` commands and the permitted parameters and options, see the *ovconfget(1)* and *ovconfchg(1)* manual pages.

Chapter 9: HPOM Service Navigator

In this Chapter

This chapter contains configuration and reference details about Service Navigator. In this chapter, you can find information about the following topics:

For more detailed information about installation requirements and instructions regarding Service Navigator, see the *HPOM Installation Guide for the Management Server*.

Service Navigator Overview

HP Operations Service Navigator is a component of the Java GUI. It enables you to manage your IT environment while focusing on the IT services you provide.

Service Navigator depends on the monitoring, message, and action capabilities HPOM provides. If a problem occurs on one of the objects managed by HPOM, a message about this problem is generated and sent to the user responsible for the area concerned. Service Navigator maps this message to the service that is affected by the problem, and sends it to the user responsible for that service.

The severity status of the problem also changes the severity status of the service so that the user can easily identify services that are in a problematic state. To solve service-related problems, HPOM's problem resolution capabilities include service-specific analysis operations and actions.

If enabled, Service Navigator logs each change of the status in the database so that reports about service availability can be generated.

For more information about Service Navigator, see *HPOM Java GUI Operator's Guide*.

Configuring Service Navigator

Service Navigator is easily configured if you follow these instructions:

1. Plan your service hierarchy.

   Identify the managed elements that are part of your service and plan your service hierarchy accordingly.

   For more information, see "Planning Your Service Hierarchy" on page 343.

2. Write the service configuration file (or convert existing configuration files to the XML syntax).

   For more information, see "Writing the Service Configuration File" on page 344. See also "Service Configuration File" on page 364 for more detailed instructions for writing the service configuration file.

3. Activate your new service configuration with `opcservice`.

   For more information, see "Activating the Service Configuration" on page 345.

4. Set the message attribute service in HPOM. Configure HPOM so that it knows how to assign messages to services.

   For more information, see "Setting the Service Name in HPOM" on page 346.

5. Plan the operator responsibilities and assign them with `opcservice`.

   For more information, see "Assigning Services to Operators" on page 346.

6. Plan your service hours.

   For more information, see "Planning Your Service Hours" on page 348.

7. Plan the reports you need for each service and enable service logging accordingly.

   For more information, see "Enabling Service Logging" on page 348.

8. Enable and configure service status calculation view(s).

   For more information, see "Enabling and Configuring Service Multi-status Calculation" on page 349.

9. Monitor service multi-status changes.

   For more information, see "Monitoring Service Multi-status Changes" on page 351.

10. Set service attributes.

    For more information, see "Setting Service Attributes Dynamically" on page 352 and "Setting Service Attributes" on page 371.

11. Label service icons if needed.

    For more information, see "Labeling Service Icons" on page 362.

Note: Some HP Operations Smart Plug-ins provide out-of-the-box service instrumentations.

Planning Your Service Hierarchy

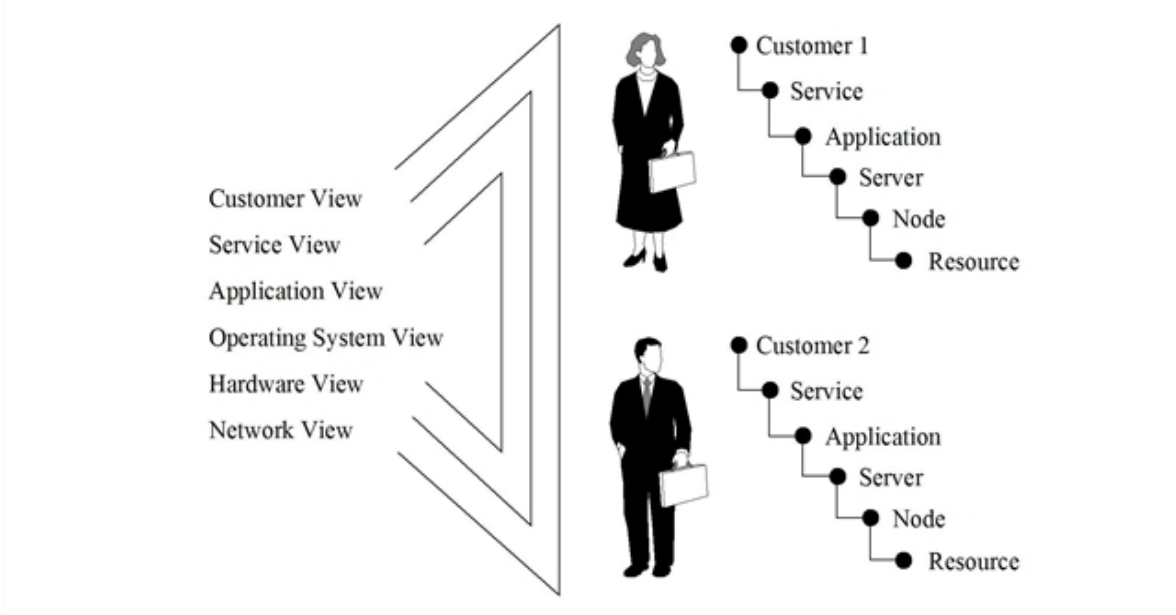
We recommend that you draft your service hierarchy before you start writing the service configuration file. When planning your service hierarchy, keep the following questions in mind:

- Which IT services do you provide? Which ones do you want to monitor?
- Who are the customers of your services? Which organizations, departments, or lines of business?
- How can you logically group the services you provide? Which services are used by other services?
- How do problems in one service affect other services? Which status propagation rule should you apply?
- How do you evaluate the severity of a problem? Which status calculation rule should you apply?
- Which service actions should be assigned to each service?

For example, you may want to organize your service hierarchy as illustrated in Figure 12: your service hierarchy could start with the customers for which you provide services, followed by the business services you provide for them (for example, a backup service). The next level could be the application level. In the case of your backup service, this could be the HP Data Protector application you are using for backup purposes at your customer sites. The following level could be the software or operating system level where you monitor the health of the operating system installed on your backup servers. The next level could be the hardware level where your servers and clients are monitored. Finally, the last level could be the network level which enables you to control the health of all network components.

Note: Only ASCII characters are allowed in service names.

Figure 12: Service Hierarchy Example



For more information about planning a service hierarchy, see also "Naming Schema for Services" on page 395.

You can also have a look at the example service configuration files. They show you what a service hierarchy can look like and how to use the XML tags. The example configuration files are available in the following directory on the management server after the installation of Service Navigator:

```
/opt/OV/OpC/examples/services/
```

For more information about each file, see the `README` file. For more information about example configuration files, see "Using Example Configuration Files" on page 397.

Writing the Service Configuration File

When you have a good plan of your service hierarchy, you can start putting it into practice by writing the service configuration file. In general, you have the choice between the following methods:

- **Manually**

  Manually write (or edit) a service configuration file as explained later in this section.
- **Automatically**

  Automatically generate a service configuration file using scripts or programs. Service Navigator provides some shell scripts in the directory `/opt/OV/OpC/examples/services/` that help you to get started.

Use the example configuration files to copy the sections of the configuration syntax that you need. For information about how parts of the SAP/R3 service hierarchy are configured, see "Service Configuration File" on page 364.

Consider the following tips when writing the service configuration file:

- **File Name Extensions**

  Save the service configuration files with the standard file name extension `.xml`.
- **Large Service Hierarchies**

  If you are planning a large service hierarchy, it can be useful to distribute the service configuration over multiple configuration files. This would make it easier for you to maintain your configuration. For example, you could define your business services and application services in one file, and all hardware and software services in another file.

Writing the Service Configuration File Manually

To write the service configuration file manually, follow these steps:

1. Change to the directory that contains the example configuration files. Enter the following:

   ```
   cd /opt/OV/OpC/examples/services/
   ```
2. As user `root`, copy the file that comes closest to your planned hierarchy to a temporary directory.

Enter the following:

```
cp <service_example> /tmp <service_config>
```

3. Open the copied example file using an XML or a plain text editor of your choice and translate the service hierarchy you have planned into the service configuration syntax. For more information, see "Service Configuration File Syntax" on page 373.

   Configuring a service includes the following aspects:

   a. Defining the services and subservices, and the relationships between them.
   b. Defining the status calculation and status propagation rules.
   c. Defining the service-specific actions.
   d. Defining the service attributes.
   e. Defining operator assignments.

4. Use the `opcservice` command on the completed configuration file to ensure that your changes are correct:

   ```
   /opt/OV/bin/OpC/opcservice -check /tmp/<service_config>
   ```

   If the syntax of the configuration file is correct, `opcservice` outputs an OK message. If the configuration file contains any errors, errors messages are displayed. Fix any syntax errors before continuing.

   If element values for service files are not specified, warning messages are displayed. For details about handling error messages, see "Service Files Error Checking" on page 346.

Activating the Service Configuration

When you finish writing the service configuration file, use the command `opcservice` to load the new configuration. Enter the following:

```
/opt/OV/bin/OpC/opcservice -add /tmp/<service_config>
```

`opcservice` passes the configuration to the `opcsvcm` process which maintains the following internal service repository file:

```
/etc/opt/OV/share/conf/OpC/mgmt_sv/opcsvcm/services
```

Note: You should not edit this file directly. Any changes done during runtime are lost the next time the `opcsvcm` process rewrites the repository file.

Modifying the Service Configuration

If you need to modify the configuration, execute the following with the management server running:

```
opcservice -list -all -xml > current_ServNav.xml
cp current_ServNav.xml new_ServNav.xml
```

Edit the new configuration file, remove the `<Results>` and `</Results>` tags and update the configuration with the following command:

```
opcservice -add new_ServNav.xml
```

> **Note:** Do not use the **`-replace`** option, otherwise you will have to re-do the operator assignments.
>
> The execution of **`opcservice -list -all -xml`** is time consuming.

You can also get an overview of your configuration by listing all services with **`opcservice -list -all`**.

For more information about available command line options for `opcservice`, see "opcservice Command" on page 396.

Service Files Error Checking

HPOM implements error checking of service files. If elements values are not specified, by default warning messages are displayed. For example:

```
Warning: Operation 'Add' : Element 'car_manufact' Label is empty,
using Element name instead. (SVC50-3) (SVC10-123)

Warning: Operation 'Add' : Element 'supply-chain' Label is empty,
using Element name instead. (SVC50-3) (SVC10-123)
```

To enable error checking, enter the following command:

```
ovconfchg -ovrg server -ns opc -set OPC_SVCM_ADD_WARN_IF_EXISTS <mode>
```

Where *<mode>* is one of the following:

- `TRUE`
- `FALSE`

Setting the Service Name in HPOM

When you have set up your services, you need to tell HPOM how to match messages against services. You can provide the service name by specifying it as a parameter of the `opcmsg(1)` command.

For more information, see the *opcmsg(1)* manual page.

Assigning Services to Operators

A service is visible only to the operator or operators to whom the responsibility for that service is assigned. If a service is assigned to more than one operator, its status is visible to all operators.

The operators should also have message groups and node groups assigned. If responsibilities for services, nodes, and message groups do not overlap, the operators can receive messages from a service but do not have the corresponding nodes in their responsibility matrix. Messages from assigned services are also displayed in the Java GUI so that an operator who uses the Java GUI also sees messages from assigned services.

Operator capabilities are not restricted on service messages or on the nodes from which service messages are received. For example, if an operator is in general not allowed to own or disown messages, the messages from assigned services can still be owned or disowned. Similarly, actions can be executed on service nodes, even if these nodes are not in the responsibility matrix of the operator.

Services can be assigned using one of the following methods:

- **opcservice command**

  Use the `opcservice` command to assign services to operators. This is the recommended method.
- **service configuration file**

  Define the assignment in the service configuration file using the `<Operator>` tag.

If you want to find out which operators are responsible for your services, use `opcservice` with the `-operators` option. This outputs a list of all operators who have services assigned as well as the assigned services. Service assignments are kept in the HPOM database.

Assigning Services with opcservice

To assign services with the `opcservice` command, perform the following:

1. Decide which operator is responsible for a service.

   Note that in a complex service hierarchy it may be useful to assign subservices to different operators instead of the full hierarchy.
2. Use `opcservice` to assign the service and all contained subservices to the operator. Enter the following:

   `/opt/OV/bin/OpC/opcservice -assign <operator> <service_name>...`

   Where are:

<operator>	HPOM logon name of the operator.
<service_name>	Name of the service as defined in the service configuration file. Multiple service names can be listed.

   For more information about available command line options for `opcservice` (for example for deassigning services from operators), see also "opcservice Command" on page 396.

Assigning Services to User Profiles

Besides assigning services to each operator separately, it is possible to assign services to user profiles. To do so, run the following command:

```
/opt/OV/bin/OpC/opcservice -assign <profile> <service>
```

Note: To ensure that new assignments of user profiles are reflected in the currently running Java GUI sessions, select **File > Reload Configuration**.

Assigning Services in the Configuration File

Service assignments can also be specified in the service configuration file. This involves defining the operators with the XML tag `<Operator>` and specifying the preferred services.

This method has the advantage in a possibility to define additional aspects of the Service Navigator GUI (for example, you can define the label, description, or icon of the topmost service assigned to the operator). Using `opcservice` is the recommended method. For more information, see also "Setting up Service Assignments" on page 372.

Planning Your Service Hours

With HPOM you can set up service hours and schedule outages. Messages which arrive outside of defined service hours are buffered, and messages which arrive during a scheduled outage are suppressed. For more information, see "Flexible Management Configuration" on page 89. For more details see also the *HPOM Concepts Guide*.

Enabling Service Logging

If enabled, Service Navigator keeps a log of each status change in the HPOM database. This enables you to design and generate reports about the availability of your services based on the data kept in the HPOM database. Perform the following:

1. Decide for which services you want to enable logging.
2. Enable logging by entering the following:

   ```
   /opt/OV/bin/OpC/opcservice -log_enable <service_name>
   ```

   To enable logging for all subservices or for subservices up to a certain level, use the `-recursive` or `-depth` options respectively.
3. To check for which services logging is enabled, enter the following:

   ```
   /opt/OV/bin/OpC/opcservice -logs
   ```

For more information about disabling service logging and other command line option, see also the *opcservice(1m*) manual page.

The service logging file lists the services for which logging is enabled. It is located in:

`/etc/opt/OV/share/conf/OpC/mgmt_sv/opcsvcm/loggings`

Note: Never edit this file directly.

Use the command line tools `opcsvcdwn` and `opcsvcupld` to download or upload service logs from or into the HPOM database. Downloading service logs can become necessary when too many logs exist in the HPOM database and the amount of available disk space is not sufficient. For more information about these commands, see also the *opcsvcdwn(1m)* and *opcsvcupld(1m)* manual pages. More details about service logging and the service-related tables are also available in the *HPOM Reporting and Database Schema*.

Enabling and Configuring Service Multi-status Calculation

You can specify service status calculation views that you want to be enabled on your HP Operations management server, as well as set the global default service status calculation view.

To enable and configure multi-status calculation, perform the following:

1. Log on as user `root` to the HP Operations management server.
2. Enable or disable service status calculation views on your HP Operations management server. Enter the following:

   `ovconfchg -ovrg server -ns opc -set JGUI_MULTISTATUS_ENABLED <calc_value>`

   Where *<calc_value>* is one of the following:

   `TRUE` (to enable service multi-status calculation)

   `FALSE` (to disable service multi-status calculation)

   Caution: You must enable at least one service status calculation view. If you fail to do so, Service Navigator will fail at its startup.

   - To enable or disable the `Overall` calculation view, enter the following:

     `ovconfchg -ovrg server -ns opc -set OPC_SVC_CALC0 <calc_value>`

     Where *<calc_value>* is one of the following:

     `TRUE` (to enable this status view)

     `FALSE` (to disable this status view)

   - To enable or disable the `Operational` calculation view, enter the following:

     `ovconfchg -ovrg server -ns opc -set OPC_SVC_CALC1 <calc_value>`

Where *<calc_value>* is one of the following:

TRUE (to enable this status view)

FALSE (to disable this status view)

For example, if you want to enable the Operational calculation view, and to disable the Overall calculation view, enter the following:

```
ovconfchg -ovrg server -ns opc -set OPC_SVC_CALC0 FALSE
ovconfchg -ovrg server -ns opc -set OPC_SVC_CALC1 TRUE
```

3. Specify default service status calculation views. Enter the following:

   ```
   ovconfchg -ovrg server -ns opc -set OPC_SVC_DEFAULT_CALC <calc_value>
   ```

   Where *<parameter>* and *<calc_value>* are the service status calculation view and its value. This value can be one of the following:

   - 0 (Overall calculation view)
   - 1 (Operational calculation view)

     For example, to set your default service status calculation view to Operational, enter the following:

     ```
     ovconfchg -ovrg server -ns opc -set OPC_SVC_DEFAULT_CALC 1
     ```

Caution: You must restart the Service Navigator for changes to take effect.

Renaming Service Status Calculation Views

To rename the service status calculation views, set the following parameters using ovconfchg utility:

```
OPC_SVC_CALC_NAME0 <calc_name1>
OPC_SVC_CALC_NAME1 <calc_name2>
```

Where *<calc_name1>* and *<calc_name2>* are the names of service status calculation views.

For example, if you want to rename the Overall calculation view to Overall_1, enter the following:

```
ovconfchg -ovrg server -ns opc -set OPC_SVC_CALC_NAME0 Overall_1
```

Displaying Service Status Calculation Views in non-English Environments

In non-English environments, the values of OPC_SVC_CALC_NAME0 and OPC_SVC_CALC_NAME1 configuration variables have to be manually specified after enabling multi-status calculation views in order to be displayed in the chosen language.

For example, to have the Overall status calculation view displayed in the preferred language, enter the following:

```
/opt/OV/bin/ovconfchg -ovrg server -ns op -set OPC_SVC_CALC_NAME0 <Overall_
translated>
```

Likewise, to have the `Operational` calculation view displayed in the preferred language, enter the following:

```
/opt/OV/bin/ovconfchg -ovrg server -ns opc -set OPC_SVC_CALC_NAME1 <Operational_
translated>
```

Where *<Overall_translated>* and *<Operational_translated>* are the translations of these views in the chosen languages.

Monitoring Service Multi-status Changes

You can monitor the service multi-status changes and check the current service status calculation view in one of the following ways:

- **By using the `opcservice` command**

  To check the status calculation view for the service `perfsvc`, perform the following:

  a. Log on as user `root` to the HP Operations management server.

  b. Enter the following:

  ```
  opcservice -list perfsvc -xml
  ```

  You should get output similar to the following:

  ```
  <Results>
    <Services>
      <Service>
        <Name>perfsvc</Name>
        <Status>
          <Normal/>
        </Status>
        <MultiStatus>
          <CalculationId>1</CalculationId>
          <Normal/>
        </MultiStatus>
        <Label>Performance Service</Label>
      </Service>
      <Association>
        <Composition/>
        <SourceRef>perfapp</SourceRef>
        <TargetRef>perfsvc</TargetRef>
        <Status>
          <Normal/>
        </Status>
        <MultiStatus>
          <CalculationId>1</CalculationId>
          <Normal/>
        </MultiStatus>
  ```

```
      </Association>
    </Services>
  </Results>
```

- **By Creating Your Own Applications**

  You can create your own applications using C++ Service Engine APIs. For more information, see the *opcsvc_api(1m)* manual page.

- **By using the HPOM Java GUI**

  You can monitor service multi-status changes by using the HPOM Java GUI. See *HPOM Java GUI Operator's Guide* for more information.

Setting Service Attributes Dynamically

Service attributes are used to provide additional information about a service. You can define service attribute values in the service configuration file. Information on defining service attributes is available in "Setting Service Attributes" on page 371. Service attributes that are specified in the service configuration file are loaded at startup and cannot be updated during runtime.

Attribute values can also be added dynamically without the need to update the service configuration file and restart Service Navigator. This enables Service Navigator to be always updated with the latest service attribute values and you are always presented with the latest service information. You can also dynamically update service attribute values, which are already specified in the service configuration file. Any values that are dynamically set can also be stored and reloaded the next time Service Navigator is started by enabling the recovery feature.

There are two ways to set service attributes dynamically:

- **opcsvcattr command line tool**

  `opcsvcattr` is a shell script that can be called on the command line to add, change, or remove service attributes.

  This script is useful for testing purposes, it verifies if the service attributes are updated as expected. You can also call this tool as an automatic action for a message. Beware though that performance could be impaired.

  Logging is not available for this method.

  To find out how to use the `opcsvcattr` tool, see "Setting Service Attributes Dynamically with opcsvcattr" on page 353.

- **HPOM messages**

  HPOM message attributes can be used to convey updated information used to dynamically set service attribute values. The following message attributes are used:

  - Application and object message attributes

    This method is available for all message sources.

- Custom message attributes

  This method is only available for message sources where custom message attributes are supported: logfiles, `opcmsg(1|3)` and threshold monitor.

Using HPOM messages to set service attributes has the advantage that performance is not impaired and that the changes can be logged. However, the configuration effort is higher than when using the `opcsvcattr` tool.

For more information about this method, see "Setting Service Attributes Dynamically with HPOM Messages" on page 354.

Attributes are added at runtime and when the Service Navigator is stopped, the values are deleted unless message logging is enabled. A service attribute that is added dynamically overrides the value of the service attribute of the same name. When the service configuration is reloaded, the service attribute value that is defined in the service configuration file will again be displayed unless message logging is enabled.

To find out how to enable logging, see "Message Logging for opccustproc1" on page 358.

Tip: You can save a current service configuration to a file in XML format with the **`opcservice -dump <filename>`** command. Portions of this configuration can be used to enhance the associated service configuration file.

Setting Service Attributes Dynamically with opcsvcattr

Use the shell script `opcsvcattr` to add, change, or remove service attributes dynamically. For example, to add the three labels, `Master`, `Low`, and `10.5V`, to the service `Error`, enter the following command on the management server:

```
/opt/OV/bin/OpC/opcsvcattr svc_id=Error name=ov_label1 value=Master name=ov_label2
value=low name=ov_label3 value=”10.5V”
```

The icon representing the service `Error` now displays three new labels.

Tip: To add additional labels to service icons Service Navigator uses reserved attribute names such as `ov_label1`, `ov_label2`,... `ov_labeln`.

To learn more about labeling service icons, see "Reserved Service Attributes" on page 393 and "Labeling Service Icons" on page 362.

The `opcsvcattr` tool can be used as an automatic action in the HPOM policies. Beware though that this may cause performance problems. Ideally, `opcsvcattr` should only be used when testing the configuration of service attributes.

Syntax for `opcsvcattr`:

opcsvcattr	svc_id=<*svc_name*> {name=<*name*> value=<*value*>}|
	-remove svc_id=<*svc_name*> {name=<*name*>}|
	-removeall svc_id=<*svc_name*>|
	-removepref svc_id=<*svc_name*>name=<*name_prefix*>,

In this instance:

<svc_name>	Name of the service to which the service attributes apply.
<*name*>	Service attribute name. For example ov_label1.
<*value*>	Service attribute value to be displayed.
-remove	Removes all specified service attributes.
-removeall	Removes all service attributes.
-removepref	Removes all service attributes which name matches the specified name prefix.
<*name_ prefix*>	Prefix string for -removepref.

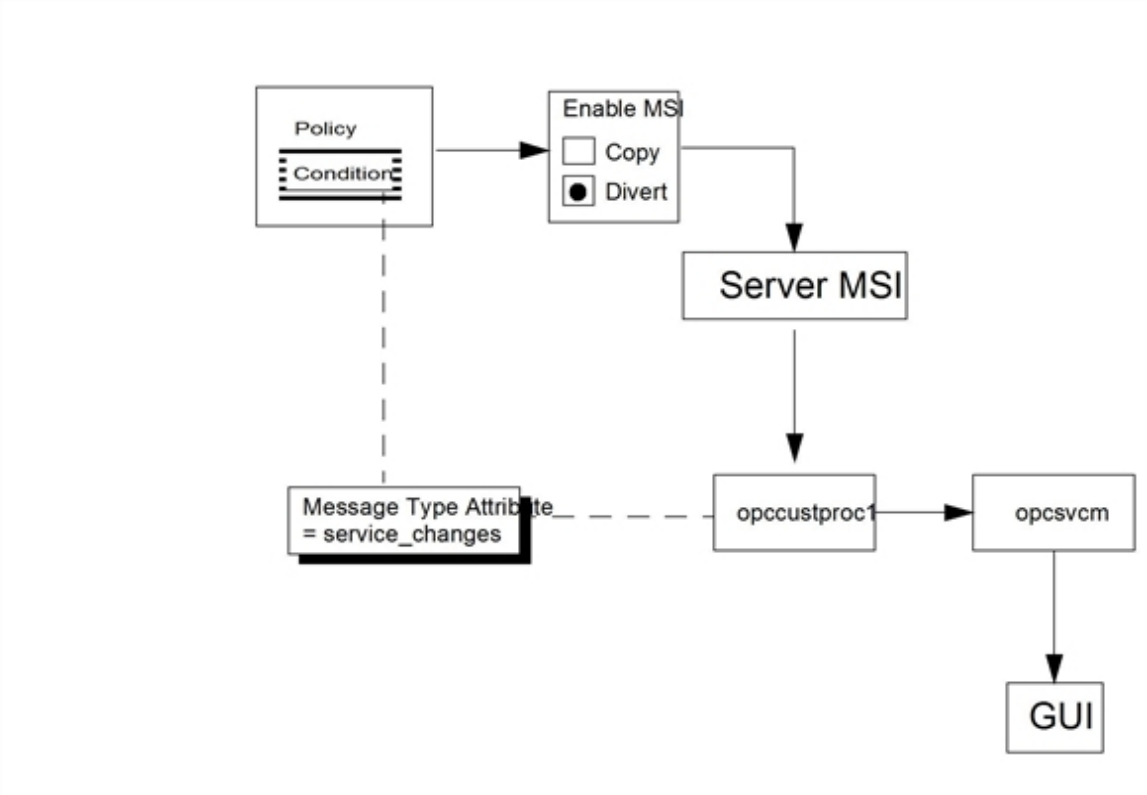
For more information about opcsvcattr, see the *opcsvcattr(1m)* manual page.

Setting Service Attributes Dynamically with HPOM Messages

Use HPOM messages to transport service attributes from managed nodes to the management server. Messages that contain information relevant for service attributes are fed into the server message stream interface, the message flow of the management server. On the management server, the process opccustproc1 retrieves those messages from the flow and, using the opcsvcm process, changes or removes the service attributes in the Service Navigator GUI.

For an illustration of the message flow from the managed nodes to the Service Navigator GUI through the management server, see Figure 13.

Figure 13: Message Flow with opccustproc1



Information for service attributes can be attached in two ways to HPOM messages, both are available for all message sources. For these methods, the following message attributes are used to convey relevant information:

- Application and object message attributes
- Custom message attributes

Steps for Setting Service Attributes Dynamically with HPOM Messages

To set service attributes dynamically with HPOM messages, follow the procedure:

1. On the management server, enable the `opccustproc1` process:
   a. Move the `opccustproc1` file from the following location:
   `/opt/OV/contrib/OpC/opccustproc`
   to this one:
   `/opt/OV/bin/OpC`
   b. Move the `libopccustproc1.so` library from the following location:
   `/opt/OV/contrib/OpC/opccustproc`
   to one of the following locations, depending on your system:

- HP-UX [IPF32]: `/opt/OV/lib/hpux32`
- Solaris: `/opt/OV/lib`
- Linux x86_64: `/opt/OV/lib64`

c. Register the `opccustproc1` process with `ovc` by entering the following command:

```
/opt/OV/bin/ovcreg -add /opt/OV/contrib/OpC/opccustproc/opccustproc1.xml
```

d. Restart the management server by entering the following:

```
/opt/OV/bin/OpC/opcsv -start
```

This integrates `opccustproc1` into the management server processes.

Note: The `opccustproc1` process can be controlled with the `ovc` or `opcsv` commands. For example, to start and stop only the `opccustproc1` process by using `ovc`, run the following commands:

```
ovc -start opccustproc1
ovc -stop opccustproc1
```

2. On the management server, enable output to the server message stream interface (MSI):

```
/opt/OV/bin/OpC/opcsrvconfig -msi -enable
```

(Do not send all messages to the server MSI.)

3. Prepare a policy and condition with the following characteristics:

a. In policy body, use keyword `MPI_SV_DIVERT_MSG` to divert messages to Server MSI. You can use `MPI_SV_COPY_MSG` to copy them instead, but consider that the message browser can become filled with irrelevant messages.

Divert the messages rather than copying them to avoid filling the message browser with irrelevant messages.

b. Set the message type attribute to `service_changes` by using the keyword `MSGTYPE`.

`opccustproc1` uses the message type attribute to identify the messages to be sent to Service Navigator.

c. Set custom message attributes or the application and object message attributes:

- Application and object message attributes

  Use keywords `APPLICATION` and `OBJECT` in the `SET` section of the policy body to set the values of application and object fields of the message.

- Custom message attributes

  You need to define two attributes: the first attribute specifies the operation (set or remove), the second attribute specifies the parameters (service attribute name and value). Use keyword `CUSTOM` followed by the name and the value of the preferred attribute.

For detailed information about the required syntax, see "Syntax for Setting Service Attributes Dynamically" on page 357.

4. Distribute the policy to the managed nodes.

Syntax for Setting Service Attributes Dynamically

When setting service attributes with `opccustproc1`, you always need to define two parts:

- **Operation**

  The operation part instructs `opccustproc1` what to do: set or remove a service attribute.

  When setting service attributes with custom message attributes, the name of the custom message attribute is `OV_OPERATION`.

- **Parameter**

  The parameter part contains the data for the service attribute. It consists of a name and the corresponding value.

  When setting service attributes with custom message attributes, the name of the custom message attribute is `OV_PARAMS`.

Table 22 describes the syntax for custom message attributes. The `OV_OPERATION` column lists the operations that are possible (set or remove). The `OV_PARAMS` column lists the names and values for service attributes.

When setting service attributes with application and object message attributes, enter the operation (set or remove) in the `Application` field, and the parameters (service attribute name and value) in the `Object` field.

Table 22: Syntax for Custom Message Attributes

Custom Message Attribute Name	`OV_OPERATION`[a]	`OV_PARAMS`[b]
Custom Message Attribute Value	`SVC_ATTR_SET`	`[name=value] [name=value]...`
	`SVC_ATTR_REMOVE`	`[name] [name]...`
	`SVC_ATTR_REMOVE_PREF`	`[name_prefix]`
	`SVC_ATTR_REMOVE_ALL`	

In this instance:

`SVC_ATTR_SET`	Sets or adds service attributes specified in `OV_PARAMS`.

[a]Corresponds to the application message attribute.

[b]Corresponds to the object message attribute.

SVC_ATTR_REMOVE	Removes service attributes specified in OV_PARAMS.
SVC_ATTR_REMOVE_PREF	Removes service attributes, which name matches the name prefix specified in the OV_PARAMS.
SVC_ATTR_REMOVE_ALL	Removes all service attributes. OV_PARAMS custom message attributes are ignored.

Message Logging for opccustproc1

Service attributes that are added dynamically are not stored in the database and are therefore lost after the Service Engine opcsvcm process is restarted. Dynamic service-related data can be logged and reused by configuring the opccustproc1 process.

Note: This is not possible for service attributes that are added with opcsvcattr.

The format of the logfile is XML. The message ID, the service ID, OV_OPERATION and OV_PARAMS data are stored from each message. The file is stored on the management server at the following location:

```
/var/opt/OV/share/tmp/OpC/mgmt_sv/opccustproc1.xml
```

When the opccustproc1 process is started as part of the ovc -start or opcsv -start commands, it reads the messages from the opccustproc1.xml logfile when they are received from the message stream interface. Once all stored messages are processed, opccustproc1 registers with the message stream interface and reverts to the normal operation.

To enable logging for the opccustproc1 process on the HP Operations management server, use the ovconfchg command line tool. Enter the following:

```
ovconfchg -ovrg server -ns opc -set OPC_CUSTPROC1_LOG TRUE
```

The default maximum number of messages that can be written to the opccustproc1 logfile is 1000. You can change this number to 5000, for example, by using the following command:

```
ovconfchg -ovrg server -ns opc -set OPC_CUSTPROC1_LOG TRUE,5000
```

After the maximum number of messages is reached, for every further message that is appended to the end of the logfile, the earliest message is removed.

Example Policy for opcmsg Message Source

Service Navigator provides the example policy opcmsg(1|3)DSA for the opcmsg(1|3) message source. Use this example to set up a message source policy for your message source.

Any opcmsg message matched by this policy is fed into the server message stream interface and sent to the management server where the opccustproc1 process retrieves the message and adds, changes, or removes the service attributes accordingly in the Service Navigator GUI.

Installing the Example Policy opcmsg(1|3)DSA

Perform the following steps to install the example policy `opcmsg(1|3)DSA`:

1. Untar the upload tree that contains the example policy:

   ```
   tar -xvf /opt/OV/OpC/examples/services/dsa_upload.tar
   ```

   The upload tree will be placed in the `/tmp/dsa` directory.
2. Upload the example policy:

   ```
   /opt/OV/bin/OpC/opccfgupld -add -index dsa.idx /tmp/dsa
   ```
3. Run `opcpolicy -list_pols` command line utility. The policy `opcmsg(1|3)DSA` is now available in the policy list.
4. Assign and distribute the policy `opcmsg(1|3)DSA` to the managed nodes.

Submitting Messages for opcmsg(1|3)DSA

Use the `opcmsg` command line tool to submit messages for this policy.

Using Custom Message Attributes:

```
opcmsg a=new_label o=my_obj msg_text=”message text” service_id=my_service -
option OV_OPERATION=SVC_ATTR_SET -option OV_PARAMS=”[ov_label1=Master] [ov_
label2=low]”
```

In this example, a message is submitted that adds two labels for the service `my_service`. The first label has the text `Master` and the second label has the text `low`. The application attribute of this message is set to `new_label` to match the attribute that is set in the policy condition.

To create a condition that would match the `new_label` value of the application field, use keyword `APPLICATION` followed by `new_label` in the `MSGCONDITIONS` block of the policy body. Setting the message type to `service_changes` is performed be entering the keyword `MSGTYPE` in the `SET` block of the policy body.

For details about policy body grammar, refer to the *HPOM Concepts Guide*.

You can find the values of the custom message attributes which belong to the `policy opcmsg (1|3)DSA` using the keyword `CUSTOM`. Look for the attributes with the following name patterns: `*OV_ OPERATION*` and `*OV_PARAMS*`.

Using Application and Object Message Attributes:

```
opcmsg a=SVC_ATTR_SET o=”[ov_label1=Master] [ov_label2=low]” msg_text="message
text" service_id=my_service
```

This example does exactly the same as the example above, but uses the application and object message attributes instead. The application attribute of this message is set to `SVC_ATTR_SET`. It is the operation type for setting the service attributes recognized by the `opccustproc1` and is simultaneously one of the match conditions specified in the `opcmsg(1|3)DSA` policy.

The following is an excerpt from the `policy opcmsg(1|3)DSA` that uses an application field of the condition block to perform the operation:

```
MSGCONDITIONS
DESCRIPTION "Application and Object"
CONDITION
APPLICATION "SVC_ATTR_SET|SVC_ATTR_REMOVE|
 SVC_ATTR_REMOVE_PREF|SVC_ATTR_REMOVE_ALL"
SET
MSGTYPE "service_changes"
```

opccustproc1 in Flexible Management Environments

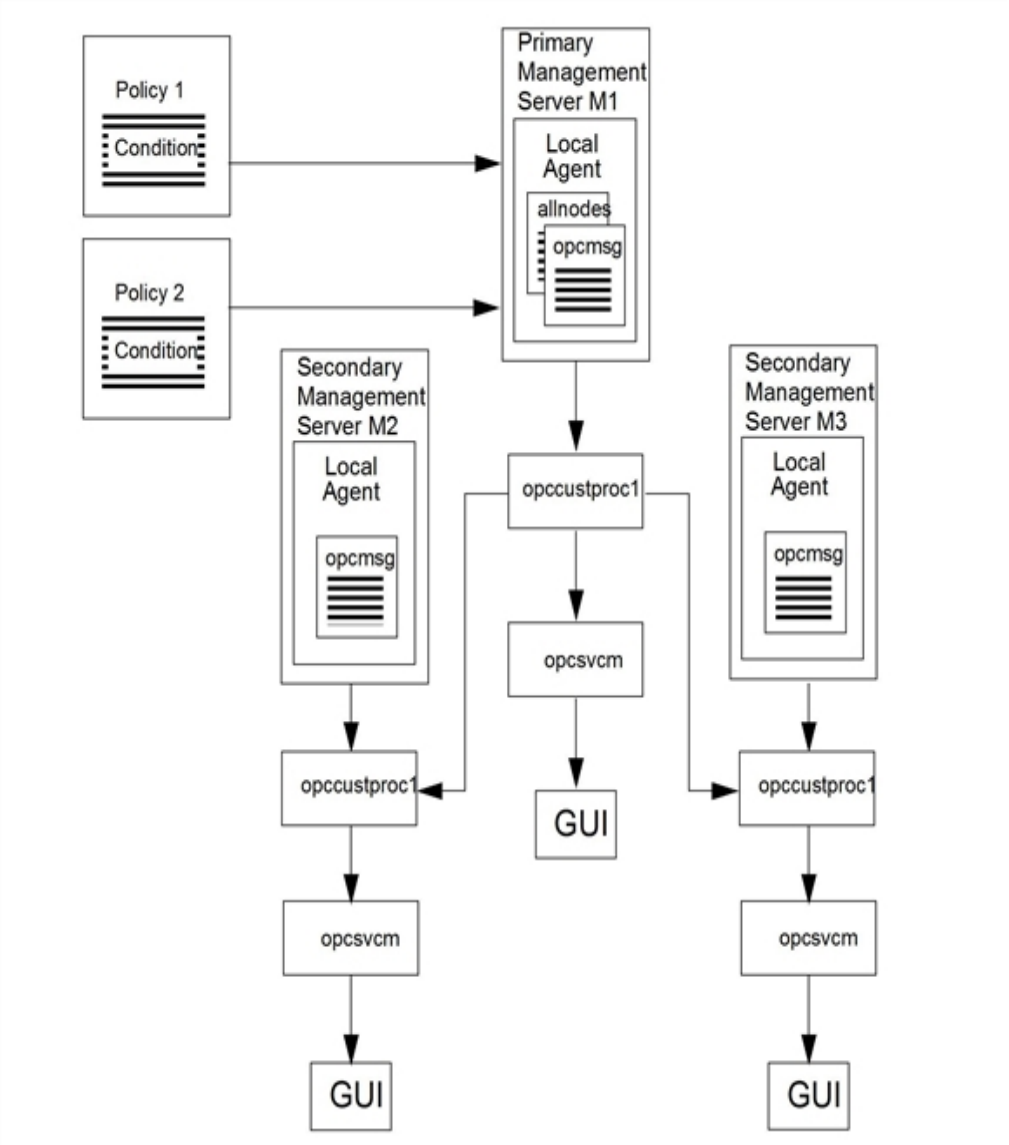
If you are operating in a flexible management environment (with one or more secondary management servers) you need to ensure that the `opccustproc1` process forwards messages from the primary management server to other management servers as defined in the flexible management configuration file `allnodes`. This file is located on the primary management server.

Note: `opccustproc1` must be running on the primary and all secondary management servers.

The example policy `opcmsg(1|3)DSA` can be used for this purpose. It contains a condition `MoM_forward` that instructs the local agent of the primary management server to send matched HPOM messages to the secondary management server.

Figure 14 shows how `opccustproc1` on the primary management server M1 forwards messages to the secondary management servers M2 and M3. All management servers have `opcmsg` files installed, while only the primary server has `allnodes` file installed. It is not necessary to have the `allnodes` file installed on the secondary management servers for this purpose.

Figure 14: Message Flow with opccustproc1 in Flexible Management Environments



Configuring Flexible Management with opccustproc1

To configure flexible management with `opccustproc1`, follow the procedure:

1. Install the example policy `opcmsg(1|3)DSA` as described in "Installing the Example Policy opcmsg (1|3)DSA" on page 359.
2. Make sure that the managed nodes of your primary management server are added to the `Node Bank` and to the `Node Group Bank` of your secondary management server.
3. On the HP Operations management server, use the command-line tool `ovconfchg`. Run the following command:

   ```
   ovconfchg -ovrg server -ns opc -set OPC_CUSTPROC1_MOM_FORWARD TRUE
   ```

4. Restart the HP Operations processes as follows:

   ```
   /opt/OV/bin/ovc -start
   ```

5. Copy the flexible management configuration file to the distribution directory:

   ```
   cp /etc/opt/OV/share/conf/OpC/mgmt_sv/tmpl_respmgrs/allnodes
   /etc/opt/OV/share/conf/OpC/mgmt_sv/respmgrs
   ```

6. Edit the flexible management configuration file `/etc/opt/OV/share/conf/OpC/mgmt_sv/respmgrs/allnodes`. Replace the name of the example secondary management server (`tcbbn040.bbn.hp.com`) with the name of your secondary management server.

7. Distribute the policy `opcmsg(1|3)DSA` to the managed nodes to start a distribution of the `allnodes` file.

8. Submit a test message with `opcmsg` as described in "Submitting Messages for opcmsg(1|3)DSA " on page 359 to test whether the label changes on both the primary and the secondary management server.

Labeling Service Icons

Labels can be added to service icons and messages to display additional information about the service. A label can be text or an image in service presentations and can apply to all service information, such as graphs, custom service maps, service submaps, impacted services and root causes, and service icons in the shortcut bar.

Services can be labeled in two ways using reserved service attributes that are recognized by the GUI:

- **In the Service Configuration File**

  The values for these labels are acquired at Service Navigator startup and cannot be changed or removed during the current session. See "Setting Service Attributes" on page 371 for an explanation of specifying labels using the service configuration file. Labels displayed from service configuration data can be overridden dynamically.

- **Dynamically**

  It is also possible to continuously update the text or image of a label and display the most up-to-date information by using messages. The refresh rate is determined by that set for the message browser. The methods for automatically setting and updating service attributes are detailed in "Setting Service Attributes Dynamically" on page 352.

  The refresh interval of the GUI is 5 seconds or longer. All visible changes in the GUI are triggered by the refresh cycle, while dynamic changes cannot be faster than 5 seconds from initiation.

  The information represented in dynamic labels is usually retrieved from messages. The text or an image file name parsed from, for example, message text, can then be used in a service label in the Service Navigator GUI.

Syntax for Dynamic Labels

Dynamic labels use the concept of dynamically set service attributes to achieve an effect where the content and position of labels displayed for service icons changes dynamically. For this purpose, Service Navigator uses predefined reserved service attribute names. The following table explains this syntax:

Table 23: Key Attribute Names for Dynamic Labels t

To Change:	Label Text	Label Image
Attribute Name:	ov_label<*n*>	*ov_label<n>*
	<n> is a number that determines the position of the label below the service icon. Must be a positive integer.	
Attribute Value:	*<string>*	*ov_image:<path_to_image>*
	<string> must be a positive integer. *<path_to_image>* is the full path to the image in the file system.	

Text Labels

The service icon labels have a maximum width that is sufficient to display approximately 25 characters. If the label is too long to be fully displayed, it is cut short and appended with three dots (...). The tool-t on the label shows the complete label string.

Image Labels

Image indicators that contain graphic files instead of text can be used to represent service information in labels. These images are typically gauge graphs, pie charts, or icons, but can be any kind of image.

Valid image file formats are gif or jpeg.

If an image is to be used as a label, the syntax is of the format:

ov_image:*<path_to_image>*

The file name must be specified with the full path name or with the URL (http or ftp).

It is advisable to store the image files on the HP Operations management server in the following directory:

/opt/OV/www/htdocs/ito_op/images/

where they can be accessed without the full path specified.

It is also advisable to use logical file names to make later application easier. For example, a set of level gauge files could be called:

`gauge0.gif`, `gauge25.gif`, `gauge50.gif`, `gauge75.gif` and `gauge100.gif`.

Scenario for Setting Labels Dynamically

The following command examples describe how to set up additional labels. These labels are entered using `opcsvcattr`. To be of greatest value, these should be entered as variables through message attributes so as to update fully automatically.

The following commands show how these labels were applied. If the command is executed successfully, a check is made and an `OK` message is displayed.

```
# /opt/OV/bin/OpC/opcsvcattr svc_id="Port A" name=ov_label1 value="Messages: 4"
name=ov_label2 value="CPU 75%" name=ov_label3 value=ov_
image:/opt/OV/www/htdocs/ito_op/images/gauges/gauge75.gif
```

```
<Results>
  <OK/>
</Results>
```

```
# /opt/OV/bin/OpC/opcsvcattr svc_id="Port B" name=ov_label1 value="Messages: 7"
name=ov_label2 value="CPU 25%" name=ov_label3 value=ov_
image:/opt/OV/www/htdocs/ito_op/images/gauges/gauge25.gif
```

```
<Results>
  <OK/>
</Results>
```

Service Configuration File

This section helps you to write the service configuration file. The section contains examples of how the functionality available for Service Navigator can be translated into the configuration syntax. The configuration task can be split into the following tasks:

- "Creating the Service Hierarchy" on page 364
- "Defining the Rules" on page 367
- "Setting up Service Actions" on page 369
- "Setting Service Attributes" on page 371

Creating the Service Hierarchy

Before you start writing the service configuration file you should have a good plan of your service hierarchy and of the relationships between the services. For a list of questions you should consider, see "Planning Your Service Hierarchy" on page 343.

After your hierarchy is drafted, write the configuration file. Begin with setting up the hierarchy and defining the attributes of each service. For example, define the icon that represents the service in the GUI, the label that is displayed in the scoping pane or below the icon in the service graph, the title of the service graph window or of the submap displayed in the content area, and so on.

The following example shows how you can build the service hierarchy for the SAP/R3 service, from the top-level SAP/R3 service to the ABAP subservice. The service Application Server 01 contains its subservices, and the service Application Server 02 uses the OS subservice contained in Application Server 01. The name of this relationship is displayed in the GUI as `use`, and the XML tag `<Label>` is used.

The configuration file is available as an example file in the following directory on the management server:

`/opt/OV/OpC/examples/services/sap.xml`

For more information about the service tags, see the following tables:

- "Root Service Tags " on page 380
- "Service Tags " on page 380
- "HPOM Service Navigator" on page 341
- "Association Tags " on page 384

```
<?xml version=1.0 ?>
<!DOCTYPE svcengine SYSTEM service.dtd>

<Services>
  <Service>
    <Name>SAP</Name>
    <Label>SAP R/3</Label>
    <Title>SAP R/3 Service</Title>
    <Source>
      <Composition/>
      <ServiceRef>application_server</ServiceRef>
    </Source>
  </Service>
  <Service>
    <Name>application_server</Name>
    <Label>SAP Application Server Class</Label>
    <Title>SAP Application Server Class</Title>
    <Icon>/opt/OV/www/htdocs/ito_op/images/server.32.gif</Icon>
    <Source>
      <Composition/>
      <ServiceRef>application_server_01</ServiceRef>
    </Source>
  </Service>
  <Service>
    <Name>application_server_01</Name>
    <Label>Application Server 01</Label>
    <Description>SAP/R3 Application Server01</Description>
    <Source>
      <Composition/>
      <ServiceRef>ABAP</ServiceRef>
    </Source>
    <Source>
      <Composition/>
      <ServiceRef>buffer_space</ServiceRef>
    </Source>
    <Source>
      <Composition/>
      <ServiceRef>processes</ServiceRef>
    </Source>
    <Source>
      <Composition/>
      <ServiceRef>display_queue</ServiceRef>
    </Source>
    <Source>
      <Composition/>
      <ServiceRef>OS</ServiceRef>
    </Source>
  </Service>
  <Service>
    <Name>application_server_02</Name>
    <Label>Application Server 02</Label>
    <Description>SAP/R3 Application Server 02</Description>
    <Source>
      <Dependency/>
      <ServiceRef>OS</ServiceRef>
      <Label>use</Label>
    </Source>
  </Service>

</Services>
```

The previous example uses the `<Source>` tag to establish the service hierarchy. You can also use the `<Association>` tag to achieve a similar effect, the choice depends on your XML coding style: the `<Source>` tag produces nested XML, while the `<Association>` tag produces linked XML.

Caution: Make sure that all services to which you refer in your `<Association>` tags are specified in the same XML file.

Using the *<Source>* tag:

```
<Service>
   <Name>application_server_01</Name>
```

```
   <Source>
      <Composition />
      <ServiceRef>ABAP</ServiceRef>
   </Source>
</Service>

<Service>
   <Name>ABAP</Name>
</Service>
```

Using the *<Association>* tag:

```
<Service>
   <Name>application_server_01</Name>
</Service>

<Service>
   <Name>ABAP</Name>
</Service>

<Association>
   <Composition />
   <SourceRef>ABAP</SourceRef>
   <TargetRef>application_server_01</TargetRef>
</Association>
```

Defining the Rules

After you have the structure of your hierarchy in place, you can think about the status propagation and status calculation rules, as well as any weighting factors you may want to apply. For more information about the general concepts behind status propagation and calculation, and about weighting, see HPOM Java GUI Operator's Guide.

The following example is similar to the one used in "Creating the Service Hierarchy" on page 364. Some subservices have been omitted to keep the structure simple and clear, and the rules by which Service Navigator calculates the severity statuses are added.

The rules are configured as shared rules, which means that they are defined once and used wherever required. This saves you time and ensures consistency, and when you want to change a rule, you need to change the definition only once.

- **Status propagation rules**

  Status propagation rules for the subservices ABAP and OS are defined as increase. In addition, OS is weighted by the factor 2, and a message factor of 2 is defined for the Application Server 01.

- **Status calculation rules**

  The status calculation rule for the service Application Server 01 defines multiple thresholds.

For a detailed list of the calculation and propagation rule tags, see also "Calculation Rule Tags " on page 386 and "Propagation Rule Tags " on page 387.

The configuration file is available as an example file in the following directory on the management server:

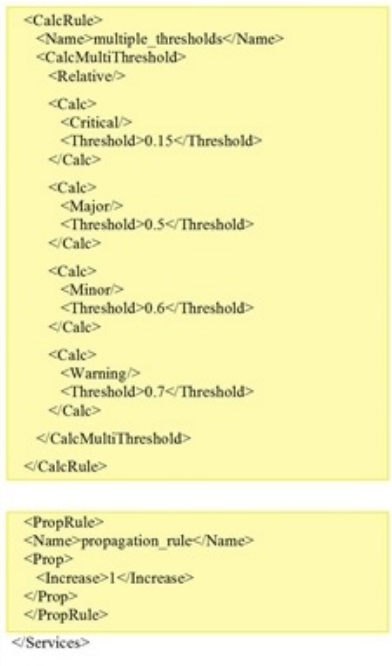
`/opt/OV/OpC/examples/services/sap.xml`

```
<?xml version=1.0 ?>
<!DOCTYPE svcengine SYSTEM service.dtd>
<Services>
  <Service>
    <Name>SAP</Name>
    <Label>SAP R/3</Label>
    <Title>SAP R/3 Service</Title>
    <Source>
      <Composition/>
      <ServiceRef>application_server</ServiceRef>
    </Source>
  </Service>

  <Service>
    <Name>application_server</Name>
    <Label>SAP Application Server Class</Label>
    <Title>SAP Application Server Class</Title>
    <Icon>/opt/OV/www/htdocs/ito_op/images/server.32.gif</Icon>
    <Source>
      <Composition/>
      <ServiceRef>application_server_01</ServiceRef>
    </Source>
  </Service>

  <Service>
    <Name>application_server_01</Name>
    <Label>Application Server 01</Label>
    <Description>SAP/R3 Application Server 01</Description>
    <MsgWeight>2</MsgWeight>
    <CalcRuleRef>multiple_thresholds</CalcRuleRef>
    <Source>
      <Composition/>
      <ServiceRef>ABAP</ServiceRef>
      <PropRuleRef>propagation_rule</PropRuleRef>
    </Source>
    <Source>
      <Composition/>
      <ServiceRef>OS</ServiceRef>
      <PropRuleRef>propagation_rule</PropRuleRef>
    </Source>
  </Service>

  <Service>
    <Name>application_server_02</Name>
    <Label>Application Server 02</Label>
    <Description>SAP/R3 Application Server 02</Description>
    <Source>
      <Dependency/>
      <ServiceRef>OS</ServiceRef>
      <Label>use</Label>
      <Weight>2</Weight>
    </Source>
  </Service>
```



Setting up Service Actions

The examples in the previous sections show you how to configure Service Navigator to display parts of the SAP/R3 service hierarchy. To allow operators to use Service Navigator fully, you can also set up service-specific actions that are executed on the node or the application that generated the message.

The attributes of a service action include a label which is displayed in the GUI, and a URL or a command that is executed. These attributes can be further modified by parameters such as: description, server, service, node, or user. For details, see "Service Configuration File Syntax" on page 373.

To make these parameters as generic as possible, use HPOM variables in the command string when defining service actions.

For example, you want to define an action with the label `Ping Node` which executes the command `/etc/ping` on the system where your service is running. Enter the following in the service configuration file:

```
<?xml version=’1.0’ ?>
<!DOCTYPE Services SYSTEM “service.dtd”>
<Services>
   <Action>
       <Name>ping</Name>
       <Label>Ping Node</Label>
       <Description>Ping IP addresses</Description>
       <Program>
          <Command>/etc/ping $OPC_SERVICE_VALUE[ip]
          </Command>
```

```
        </Program>
    </Action>
</Services>
```

For a detailed list of the service action tags, see also "Service Action Tags " on page 390.

Using Attributes in Service Actions

To check if the IP addresses of the systems where your services are running, specify them with the tag `<Attribute>` when defining the service in the service configuration file:

```
<?xml version=’1.0’ ?>
<!DOCTYPE Services SYSTEM “service.dtd”>
<Services>
<Service>
   <Name>application_server</Name>
   <Label>SAP Application Server Class</Label>
   <Title>SAP Application Server Class</Title>
   <Icon>/opt/OV/www/htdocs/ito_op/images/server.32.gif
   </Icon>
   <Attribute>
      <Name>ip</Name>
      <Value>70.154.198.255</Value>
   </Attribute>
   <ActionRef>ping</ActionRef>
   <Source>
      <Composition/>
      <ServiceRef>application_server_01</ServiceRef>
   </Source>
<Service>
</Services>
```

Setting up Local Actions

Service actions can also be executed locally on the client where the Service Navigator GUI is running. For example, to find out the hostname of the client where the Service Navigator GUI is currently running, set up the following action (it executes the command `hostname` on the Service Navigator GUI client).

```
<?xml version=’1.0’ ?>
<!DOCTYPE Services SYSTEM “service.dtd”>
<Services>
   <Action>
       <Name>Hostname</Name>
       <Label>Local Action</Label>
       <Description>Get hostnames</Description>
       <Program>
          <Command>hostname</Command>
          <Localnode/>
```

```
      </Program>
   </Action>
</Services>
```

Setting Service Attributes

Service attributes are characteristics of a service that can be defined in the service configuration file. They are used to carry additional information about a service.

Service Navigator can also handle attributes dynamically, as described in "Setting Service Attributes Dynamically" on page 352. These attributes are added at runtime and can override the values of attributes of the same name defined in the service configuration file. Updating attribute values at runtime can create dynamic effects in the Service Navigator GUI, for example changing labels of a service icon.

The following XML file shows you how you can change the appearance of a service submap with service attributes in a service configuration file. See *HPOM Java GUI Operator's Guide* for details.

The XML shown below is extracted from the example configuration file `/opt/OV/OpC/examples/services/italy_geo.xml`. In this example, the attributes of the service `Italy Banana HQ` define a map of Italy as a service submap background. This map has a width of 722 pixels and a height of 792 pixels. The subservice `Friuli branch` is positioned at 385 by 85 pixels on this background.

```
<?xml version="1.0"?>
<Services xmlns="http://www.hp.com/OV/opcsvc">
   <Service>
      <Name>italy_geo</Name>
      <Label>Italy Banana HQ</Label>
      <Icon>/opt/OV/www/htdocs/ito_op/images/eye.gif</Icon>
      <Attribute>
         <Name>ov_background</Name>
         <Value>italy.jpg</Value>
      </Attribute>
      <Attribute>
         <Name>ov_map_width</Name>
         <Value>722</Value>
      </Attribute>
      <Attribute>
         <Name>ov_map_height</Name>
         <Value>792</Value>
      </Attribute>
      <Source>
         <Composition/>
         <ServiceRef>Friuli branch</ServiceRef>
      </Source>
</Service>
<Service>
```

```
   <Name>Friuli branch</Name>
   <Label>Friuli branch</Label>
   <Icon>/opt/OV/www/htdocs/ito_op/images/banana.gif</Icon>
   <Attribute>
      <Name>ov_posX</Name>
      <Value>385</Value>
   </Attribute>
   <Attribute>
      <Name>ov_posY</Name>
      <Value>85</Value>
   </Attribute>
</Service>
```

When you have set your service attributes in the XML configuration file, use the command `opcservice` to load the new configuration. For more information, see "opcservice Command" on page 396 and the *opcservice(1m)* manual page.

Setting up Service Assignments

The last step in configuring Service Navigator is the operator-to-service assignment. If you do not want to do this with the `opcservice` command, you can define the assignments in the service configuration file. Using `opcservice` is however the recommended way.

The following example shows how to assign the service `SAP` to the operator `ito_op`. The advantage of this method is in that you can customize the Service Navigator GUI for each operator by defining attributes of the services for which the operator is responsible. For a detailed list of the operator tags, see also "HPOM Service Navigator" on page 341.

The configuration file is available as an example file in the following directory on the management server:

`/opt/OV/OpC/examples/services/sap.xml`

```
<?xml version=1.0 ?>
<?xml version=1.0 ?>
<!DOCTYPE svcengine SYSTEM service.dtd>
<Services>
  <Service>
    <Name>SAP</Name>
    <Label>SAP R/3</Label>
    <Title>SAP R/3 Service</Title>
    <Source>
      <Composition/>
      <ServiceRef>application_server</ServiceRef>
    </Source>
  </Service>

  <Service>
    <Name>application_server</Name>
    <Label>SAP Application Server Class</Label>
    <Title>SAP Application Server Class</Title>
    <Icon>/opt/OV/www/htdocs/ito_op/images/server.32.gif</Icon>
    <Source>
      <Composition/>
      <ServiceRef>application_server_01</ServiceRef>
    </Source>
  </Service>

  <Service>
    <Name>application_server_01</Name>
    <Label>Application Server 01</Label>
    <Description>SAP/R3 Application Server 01</Description>
    <MsgWeight>2</MsgWeight>
    <CalcRuleRef>multiple_thresholds</CalcRuleRef>
    <Source>
      <Composition/>
      <ServiceRef>ABAP</ServiceRef>
      <PropRuleRef>propagation_rule</PropRuleRef>
    </Source>
    <Source>
      <Composition/>
      <ServiceRef>OS</ServiceRef>
      <PropRuleRef>propagation_rule</PropRuleRef>
    </Source>
  </Service>

  <Operator>
    <Name>ito_op</Name>
    <Label>ito_ops services</Label>
    <Description>Services of operator ito_op</Description>
    <ServiceRef>SAP</ServiceRef>
  </Operator>
```

Service Configuration File Syntax

The format of the service configuration file is based on the World Wide Web Consortium Extended Markup Language (XML). You can edit it by using any text or XML editor. When you finish working on this file, use the command `opcservice` to update the HPOM configuration with your additions or modifications. `opcservice` reads the service configuration file and moves it into the following directory:

`/etc/opt/OV/share/conf/OpC/mgmt_sv/opcsvcm/services`

Note: Never edit this file directly.

More information about the `opcservice` command is available in "opcservice Command" on page 396.

Defining a service configuration includes the following tasks:

- "Defining the Services and Associations" on page 379
- "Defining Status Calculation and Propagation Rules" on page 385

- "Defining Service Actions" on page 390
- "Assigning Services to Operators" on page 391

For information about the naming schema used for services when writing the service configuration file, see "Naming Schema for Services" on page 395.

Notation Used

The Document Type Definition (DTD) for the Service Navigator XML syntax is printed in this section and is also available on the management server in the following file:

```
/etc/opt/OV/share/conf/OpC/mgmt_sv/dtds/service.dtd
```

DTD is also available in the XML Schema Definition (XSD) format in the file located in the same directory. This alternative format is based on XML and therefore is easier to read with XML editors.

The following syntax rules apply:

- **Case Sensitive**

  The Service Navigator XML parser is case sensitive, so the XML tags must be specified as defined in the DTD.

- **XML Processing Instructions**

  Each XML file must start with an XML processing instruction:

  ```
  <?xml version=’1.0’ ?>
  ```

  Comments or other tags are not allowed before this instruction.

- **Codeset**

  If no codeset is defined, the default value UTF-8 is used.

  ```
  <?xml version=”1.0” encoding=”UTF-8”?>
  ```

  The following codesets can be used with Service Navigator:

Table 24: Supported Codesets

Language	Codeset	
Czech	UTF-8 UTF-16BE UTF-16LE	ISO-8859-2
Japanese		Shift_JIS
Korean		EUC-KR
Russian		ISO-8859-5
Simplified Chinese		GB2312
Traditional Chinese		Big5
Western European (for example, English, French, German, or Spanish)		ISO-8859-1 ISO-8859-15 roman8

The codesets are the standard codeset names as defined by IANA (Internet Assigned Numbers Authority).

- **Name Spaces**

  The following namespace is used by the Service Navigator service DTD:
  `http://www.hp.com/OV/opcsvc`

  Name spaces are specified within the top-level XML tag and are used to uniquely identify the XML tags. For example, a file `services.xml` should start like this:

```
<?xml version=’1.0’ ?>
<Services xmlns="http://www.hp.com/OV/opcsvc">
```

- **Comments**

  XML provides a mechanism for commenting code which has the same syntax as for HTML comments: `<!-- comment -->`.

  Comments must not occur within declarations or inside element tags.

- **Content Model Operators**

  The following content model operators occur in the DTD:

  Table 25: XML Content Model Operators

Symbol	Usage
,	Strict ordering
|	Selection

XML Content Model Operators, continued

Symbol	Usage
+	Repetition, minimum one
*	Repetition
?	Optional
()	Grouping

- **#PCDATA**

  Elements that have character content are declared as `#PCDATA`.

- **#ASCII**

  Elements that have ASCII only character content are declared as `#ASCII`.

- **EMPTY**

  Elements that are empty are declared as `EMPTY`.

```
<!-- XML DTD for service engine files and repository -->
<!-- Services is the root element                    -->

<!ELEMENT Services (CalcRule | PropRule | Action
                    | Service | Operator | Association)* >
<!ATTLIST Services xmlns   CDATA #IMPLIED
                  version CDATA #IMPLIED>

<!ENTITY % Severity "(Normal | Warning | Minor | Major | Critical)" >

<!-- Status Calculation Rules -->
<!ELEMENT CalcRule (Name,
                    (CalcMostCritical |
                     CalcSingleThreshold |
                     CalcMultiThreshold))   >

<!ELEMENT CalcMostCritical EMPTY >
<!ENTITY % ThresholdType "(Absolute | Relative)?" >

<!ELEMENT CalcSingleThreshold (%ThresholdType;,
                               Threshold, SetTo?) >
<!ELEMENT CalcMultiThreshold (%ThresholdType;,
                              Calc*) >
<!ELEMENT Calc ((Warning | Minor | Major | Critical),
                Threshold, SetTo?) >


<!-- Status Propagation Rules -->
```

```
<!ELEMENT PropRule (Name, Prop*) >

<!ELEMENT Prop ((%Severity;)?,
                (Increase  | Decrease | Unchanged | Ignore | SetTo)) >

<!-- Actions -->
<!ELEMENT Action (Name,
                  Label?,
                  Description?,
                  (Program | URL | Applet))>

<!ELEMENT Applet    (Class, Width, Height, Parameter*) >
<!ELEMENT Parameter (Name, Value) >

<!ELEMENT Program   (Command,
                     User,
                     ServiceNodes?,
                     ServerNode?,
                     LocalNode?,
                     NodeRef*) >


<!-- Service Elements -->
<!ELEMENT Service (Name,
                   Status?,
                   MultiStatus*,
                   Label?,
                   Description?,
                   Title?,
                   Icon?,
                   Depth?,
                   Background?,
                   MsgWeight?,
                   CalcRuleRef?,
                   MsgPropRuleRef?,
                   OriginalId?,
                   (NodeRef | Attribute | ActionRef | Source | MsgSvcName)* ) >

<!-- Service Attributes -->
<!ELEMENT Attribute (Name, Value) >

<!-- Associations -->
<!ELEMENT Source ((Dependency | Composition),
                  ServiceRef,
                  Status?,
                  Label?,
                  Weight?,
                  PropRuleRef?) >
```

```
<!ELEMENT Association ((Dependency | Composition | OperatorAssignment),
                       SourceRef,
                       TargetRef,
                       Status?,
                       MultiStatus*,
                       Label?,
                       Weight?,
                       PropRuleRef?) >

<!-- Operators -->
<!ELEMENT Operator (Name,
                    Status?,
                    MultiStatus*,
                    Label?,
                    Description?,
                    Title?,
                    Icon?,
                    Depth?,
                    Background?,
                    ServiceRef*)>


<!-- Basic Attribute Elements -->
<!ELEMENT Status         %Severity; >
<!ELEMENT SetTo          %Severity; >
<!ELEMENT MultiStatus    (CalculationId, Status)>
<!ELEMENT CalcRuleRef    (#PCDATA) >
<!ELEMENT MsgPropRuleRef (#PCDATA) >
<!ELEMENT ServiceRef     (#PCDATA) >
<!ELEMENT SourceRef     (#PCDATA)>
<!ELEMENT TargetRef     (#PCDATA)>
<!ELEMENT PropRuleRef    (#PCDATA) >
<!ELEMENT ActionRef      (#PCDATA) >
<!ELEMENT AttrRef        (#PCDATA) >
<!ELEMENT NodeRef        (#PCDATA) >
<!ELEMENT Name           (#PCDATA) >
<!ELEMENT Label          (#PCDATA) >
<!ELEMENT Description    (#PCDATA) >
<!ELEMENT Threshold      (#PCDATA) >
<!ELEMENT Increase       (#PCDATA) >
<!ELEMENT Decrease       (#PCDATA) >
<!ELEMENT MsgWeight      (#PCDATA) >
<!ELEMENT Weight         (#PCDATA) >
<!ELEMENT Value          (#PCDATA) >
<!ELEMENT Command        (#PCDATA) >
<!ELEMENT URL            (#PCDATA) >
<!ELEMENT Title          (#PCDATA) >
<!ELEMENT Background     (#PCDATA) >
<!ELEMENT Depth          (#PCDATA) >
```

```
<!ELEMENT Icon           (#PCDATA) >
<!ELEMENT User           (#PCDATA) >
<!ELEMENT Class          (#PCDATA) >
<!ELEMENT Width          (#PCDATA) >
<!ELEMENT Height         (#PCDATA) >
<!ELEMENT MsgSvcName     (#PCDATA) >
<!ELEMENT OriginalId     (#PCDATA) >
<!ELEMENT CalculationId  (#PCDATA)>
<!ELEMENT Absolute           EMPTY >
<!ELEMENT Relative           EMPTY >
<!ELEMENT Unchanged          EMPTY >
<!ELEMENT Ignore             EMPTY >
<!ELEMENT Normal             EMPTY >
<!ELEMENT Warning            EMPTY >
<!ELEMENT Minor              EMPTY >
<!ELEMENT Major              EMPTY >
<!ELEMENT Critical           EMPTY >
<!ELEMENT ServerNode         EMPTY >
<!ELEMENT ServiceNodes       EMPTY >
<!ELEMENT LocalNode          EMPTY >
<!ELEMENT Dependency         EMPTY >
<!ELEMENT Composition        EMPTY >
<!ELEMENT OperatorAssignment EMPTY >

<!-- EOF -->
```

The following sections describe the tags used in the service configuration file.

Defining the Services and Associations

The following are descriptions of the service tags. The definition includes the relationship of the service to other services, and the attributes of the service.

- "Root Service Tags " on page 380
- "Service Tags " on page 380
- "Source Tags " on page 383
- "Association Tags " on page 384
- "Calculation Rule Tags " on page 386
- "Propagation Rule Tags " on page 387
- "Service Action Tags " on page 390
- "Operator Tags" on page 391

Table 26: Root Service Tags

Tag	Required?	Description
<Services>	Required.	The root element. It contains the following tags: • <CalcRule> • <PropRule> • <Action> • <Service> • <Operator> • <Association>
<CalcRule>	Any number possible.	Specifies the calculation rule.
<PropRule>	Any number possible.	Specifies the propagation rule.
<Action>	Any number possible.	Specifies the action.
<Service>	Any number possible.	Specifies the service.
<Operator>	Any number possible.	Specifies the operator.
<Association>	Any number possible.	Specifies an association.

Table 27: Service Tags

Tag	Required?	Description
<Service>	Any number possible.	Specifies the service. It contains the following tags: • <Name> • <Status> • <MultiStatus> • <Label> • <Description> • <Title> • <Icon> • <Depth> • <Background> • <MsgWeight> • <CalcRuleRef>

Service Tags , continued

Tag	Required?	Description
		• <MsgPropRuleRef> • <OriginalId> • <NodeRef> • <Attribute> • <ActionRef> • <Source> • <MsgSvcName>
<Name>	Required.	Specifies the name of the service. The name must match the string that is entered in the policy, condition, or regroup condition window, or is supplied as a parameter of opcmsg(1). The name must not be longer than 254 characters, and must not start with a hyphen (-).
<Status>	Optional.	Contains the current status of the service. This tag is ignored when specified in a service configuration file. Status queries, for example with the command opcservice -xml -list, output a severity for the service.
<MultiStatus>	Optional.	Contains the current multistatus of the service. This tag is ignored when specified in a service configuration file. Similarly as status, multistatus queries, for example with the command opcservice -xml -list, output a severity for the service.
<Label>	Optional.	Specifies the label of the service. The string entered here is displayed in the Service Navigator GUI. If <Label> is not used, <Name> is displayed.
<Description>	Optional.	Specifies the description of the service. A detailed description can be useful for future reference.

Service Tags , continued

Tag	Required?	Description
<Title>	Optional.	Specifies the title of the service window in the Service Navigator GUI. If <Title> is not used, <Label> is displayed.
<Icon>	Optional.	Specifies an icon for the service.
<Depth>	Optional.	Specifies the number of hierarchical levels displayed in the Service Navigator GUI. By default, two levels are displayed: the service itself and one more level. The number must be an integer, for example, 3.
<Background>	Optional.	Specifies a background graphic for the content pane in the Service Navigator GUI. Graphics are defined through URLs or can be called from the filesystem on the management server. If <Background> is not used, a default white background is used.
<MsgWeight>	Optional.	Specifies the weight of the service's own messages against the weight of its subservices. The default value is 1. The number can be a floating value, for example, 1.8 or 2.5.
<CalcRuleRef>	Optional.	References the calculation rule (defined in <CalcRule>) to be applied to this service.
<MsgPropRuleRef>	Optional.	References the propagation rule (defined in <PropRule>) for the service's messages.
<OriginalId>	Optional.	Identifier set by HP Operations Service Configuration for Service Navigator.
<NodeRef>	Any number possible.	References one or more nodes that belong to a service.
<Attribute>	Any number possible.	Specifies name=value pairs which can be referenced by service actions. It contains the following tags: • <Name>

Service Tags , continued

Tag	Required?	Description
		• <Value> For a list of attribute names that are reserved for Service Navigator internal usage, see "Reserved Service Attributes" on page 393.
<ActionRef>	Any number possible.	Specifies a service action.
<Source>	Any number possible.	Specifies a subservice.

Table 28: Source Tags

Tag	Required?	Description
<Source>	Any number possible.	Specifies the service. It contains the following tags: • <Dependency> • <Composition> • <ServiceRef> • <Status> • <Label> • <Weight> • <PropRuleRef>
<Dependency>	Required.	Specifies a usage relationship to a subservice.
<Composition>	Required.	Specifies a containment relationship to a subservice.
<ServiceRef>	Required.	References the subservice as defined in <Service>.
<Status>	Optional.	Contains the current status of the service. This tag is ignored when specified in a service configuration file. Status queries, for example with the command opcservice -list,-xml, output a severity for the service.
<Label>	Optional.	Specifies the name of the link between the service and the subservice. The label is

Source Tags , continued

Tag	Required?	Description
		displayed in the Service Navigator GUI.
<Weight>	Optional.	Specifies the factor to be used to modify the importance of a service's impact on its parent. This can be a decimal number. The default value is 1. The number can be a floating value, for example, 1.8 or 2.5.
<PropRuleRef>	Optional.	References the propagation rule (defined in <PropRule>) for the service's messages.

Table 29: Association Tags

Tag	Required?	Description
<Association>	Any number possible.	Specifies an association of a service to another service. It contains the following tags: • <Dependency> • <Composition> • <OperatorAssignment> • <SourceRef> • <TargetRef> • <Status> • <MultiStatus> • <Label> • <Weight> • <PropRuleRef> Using the <Association> tag has an effect similar to using the <Source> tag. The choice depends on your XML coding style: the <Source> tag produces nested XML, while the <Association> tag produces linked XML.
<Dependency>	Required.	Specifies a usage relationship to a subservice.
<Composition>	Required.	Specifies a containment relationship to a subservice.

Association Tags , continued

Tag	Required?	Description
<Operator_ Assignment>	Required.	Specifies an assignment of an operator to a service.
<SourceRef>	Required.	References the source service as defined in <Service>.
<TargetRef>	Required.	References the target service as defined in <Service>.
<Status>	Optional.	Contains the current status of the association. This tag is ignored when specified in a service configuration file. Status queries, for example with the command opcservice -xml -list, output a severity for the association.
<MultiStatus>	Optional.	Contains the current multistatus of the association. This tag is ignored when specified in a service configuration file. Similarly as status, multistatus queries, for example with the command opcservice -xml -list, output a severity for the service.
<Label>	Optional.	Specifies the name of the link between the service and the subservice. The label is displayed in the Service Navigator GUI.
<Weight>	Optional.	Specifies the factor to be used to modify the importance of a service's impact on its parent. This can be a decimal number. The default value is 1. The number can be a floating value, for example, 1.8 or 2.5.
<PropRuleRef>	Optional.	References the propagation rule (defined in <PropRule>) for the service's messages.

Defining Status Calculation and Propagation Rules

The following tags define the status calculation and propagation rules used by Service Navigator.

Table 30: Calculation Rule Tags

Tag	Required?	Description
<CalcRule>	Any number possible.	Specifies a calculation rule. It contains the following tags: • <Name> • <CalcMostCritical> • <CalcSingleThreshold> • <CalcMultiThreshold>
<Name>	Required.	Specifies the name of the calculation rule.
<CalcMostCritical>	Required.	Specifies the most critical status calculation rule. This is the default.
<CalcSingleThreshold>	Required.	Specifies the single threshold calculation rule. It contains the following tags: • <Absolute> • <Relative> • <Threshold> • <SetTo>
<CalcMultiThreshold>	Required.	Specifies a multiple threshold calculation rule. It contains the following tags: • <Absolute> • <Relative> • <Calc> • <Threshold> • <SetTo>
<Absolute>	Optional.	Specifies an absolute value for <Threshold>.
<Relative>	Optional.	Specifies a percentage value for <Threshold>.
<Threshold>	Required.	Specifies the threshold value. If an absolute threshold is specified, an integer value must be used. If a relative threshold

Calculation Rule Tags , continued

Tag	Required?	Description
		is specified, a floating point value between 0.0 (0%) and 1.0 (100%) must be used.
<SetTo>	Optional.	Specifies a severity which the service adopts. Contains one of the following tags: • <Normal> • <Warning> • <Minor> • <Major> • <Critical>
<Calc>	Any number possible.	Specifies a value for each severity. It contains the following tags: • <Warning> • <Minor> • <Major> • <Critical> • <Threshold> • <SetTo>
<Normal>	Required.	Specifies the severity normal.
<Warning>	Required.	Specifies the severity warning.
<Minor>	Required.	Specifies the severity minor.
<Major>	Required.	Specifies the severity major.
<Critical>	Required.	Specifies the severity critical.

Table 31: Propagation Rule Tags

Tag	Required?	Description
<PropRule>	Any number possible.	Specifies a propagation rule. It contains the following tags: • <Name>

Propagation Rule Tags , continued

Tag	Required?	Description
		• <Prop>
<Name>	Required.	Specifies the name of the propagation rule.
<Prop>	Any number possible.	Specifies the single threshold calculation rule. It contains the following tags: • <Normal> • <Warning> • <Minor> • <Major> • <Critical>
<Normal>	Optional.	Specifies the severity normal. It contains the following tags: • <Increase> • <Decrease> • <Unchanged • <Ignore> • <SetTo>
<Warning>	Optional.	Specifies the severity warning. It contains the following tags: • <Increase> • <Decrease> • <Unchanged • <Ignore> • <SetTo>
<Minor>	Optional.	Specifies the severity minor. It contains the following tags: • <Increase> • <Decrease> • <Unchanged • <Ignore>

Propagation Rule Tags , continued

Tag	Required?	Description
		• <SetTo>
<Major>	Optional.	Specifies the severity major. It contains the following tags: • <Increase> • <Decrease> • <Unchanged • <Ignore> • <SetTo>
<Critical>	Optional.	Specifies the severity critical. It contains the following tags: • <Increase> • <Decrease> • <Unchanged • <Ignore> • <SetTo>
<Increase>	Required.	Defines the number of severity levels by which a given severity is increased.
<Decrease>	Required.	Defines the number of severity levels by which a given severity is decreased.
<Unchanged>	Required.	Defines that severity of the subservice is not changed. This is the default setting.
<Ignore>	Required.	Defines that the severity is ignored in terms of status propagation.
<SetTo>	Required.	Specifies the severity which the service adopts. Contains one of the following tags: • <Normal> • <Warning> • <Minor> • <Major>

Propagation Rule Tags , continued

Tag	Required?	Description
		• <Critical>

For more information about the concept behind calculation and propagation rules, see *HPOM Java GUI Operator's Guide*.

Defining Service Actions

The following lines define actions that can be assigned to services. Service actions are started from a pop-up menu.

Table 32: Service Action Tags

Tag	Required?	Description
<Action>	Any number possible.	Specifies a service action.
<Name>	Required.	Specifies the name of the service action.
<Label>	Optional.	Specifies the label of the service action. The label is displayed in the pop-up menu of the Service Navigator GUI.
<Description>	Optional.	Specifies the description of the service action. A detailed description can be useful for future reference.
<Program>	Required.	Specifies the command to be executed when the service action is started. You can run only the following type of actions: • output-only applications • X-applications if they are redirected to the GUI client Contains the following tags: • <Command> • <User> • <ServiceNodes> • <ServerNode> • <LocalNode> • <NodeRef

Service Action Tags , continued

Tag	Required?	Description
<URL>	Required.	Specifies the URL of the web application to be started by the service action. The application is displayed in a separate window.
<Command>	Required.	Specifies the command to be executed when the service action is started.
<User>	Required.	Specifies the user who executes the service action. Must be specified for <Program> actions.
<ServiceNodes>	Optional.	Executes the service action on the nodes defined in <Service>.
<ServerNode>	Optional.	Executes the service action on the HP Operations management server.
<LocalNode>	Optional.	Executes the service action on the workstation where the Service Navigator is running.
<NodeRef>	Any number possible.	Specifies the nodes on which the service action will be executed.

Assigning Services to Operators

The following are descriptions of the operator tags.

Table 33: Operator Tags

Tag	Required?	Description
<Operator>	Any number possible.	Specifies an operator-to-service assignment. Contains the following tags: • <Name> • <Label> • <Description> • <Title> • <Icon> • <Depth> • <Background

Operator Tags, continued

Tag	Required?	Description
		• <ServiceRef>
<Name>	Required.	Specifies the name of the operator, for example opc_op.
<Label>	Optional.	Specifies the label of the top-level service that is assigned to the operator. The label is displayed in the Service Navigator GUI.
<Description>	Optional.	Specifies the description of the service assignment. A detailed description can be useful for future reference.
<Title>	Optional.	Specifies the title of the service window in the Service Navigator GUI. If <Title> is not used, <Label> is displayed.
<Icon>	Optional.	Specifies an icon assigned for the top-level service.
<Depth>	Optional.	Specifies the number of hierarchical levels displayed in the Service Navigator GUI. By default, two levels are displayed: the service itself and one more level. The number must be an integer, for example, 3.
<Background>	Optional.	Specifies a background graphic for the content pane in the Service Navigator GUI. Graphics are defined through URLs or can be called from the filesystem on the management server. If <Background> is not used, a default white background is used.
<ServiceRef>	Any number possible.	Specifies the service (defined in <Service>) for which the operator is responsible. If <ServiceRef> is not defined, the top-level service is assigned.

Reserved Service Attributes

The following attribute names are reserved for internal use by Service Navigator. They are used to achieve special effects in the GUI, for example for enhancing service submaps and for labeling service icons dynamically.

Attribute name:	**ov_label*<n>***
Description:	Defines additional labels for service icons. *<n>* determines the position of the label below the service icon. Must be a positive integer.
Attribute value:	The value of the attribute can be one of the following: • *<string>* Specifies label text. • ov_image:*<image>* Specifies an image to be displayed as label. *<image>* must either be the full path to the image in the file system on the management server, or a URL (http or ftp).
Example:	`<Attribute>` `<Name>ov_label1</Name>` `<Value>First Label</Value>` `<Name>ov_label2</Name>` `<Value>ov_image:/opt/OV/www/htdocs/ito_op/images/second_label.gif</Value>` `</Attribute>`
Attribute name:	**ov_background**
Description:	Defines an image to be displayed on the background of the submap of a service.
Value:	*<image>* *<image>* must either be the full path to the image in the file system on the management server, or a URL (http or ftp).
Example:	`<Attribute>` `<Name>ov_background</Name>` `<Value>/etc/opt/OV/share/backgrounds/italy.gif</Value>` `</Attribute>`
Attribute	**ov_map_width**

name:	
Description:	Defines the width of a submap background in pixels.
Value:	*<string>* *<string>* must be a positive integer. The value is defined in pixels.
Example:	<Attribute> <Name>ov_map_width</Name> <Value>100</Value> </Attribute>
Attribute name:	**ov_map_height**
Description:	Defines the height of a submap background in pixels.
Value:	*<string>* *<string>* must be a positive integer. The value is defined in pixels.
Example:	<Attribute> <Name>ov_map_height</Name> <Value>100</Value> </Attribute>
Attribute name:	**ov_posX**
Description:	Defines the position of the service icon on the x axis of the submap of the parent service (in pixels).
Value:	*<string>* *<string>* must be a positive integer. The value is defined in pixels.
Example:	<Attribute> <Name>ov_posX</Name> <Value>50</Value> </Attribute>
Attribute name:	**ov_posY**
Description:	Defines the position of the service icon on the y axis of the submap of the parent service (in pixels).

Value:	*<string>* *<string>* must be a positive integer. The value is defined in pixels.
Example:	<Attribute> <Name>ov_posX</Name> <Value>50</Value> </Attribute>

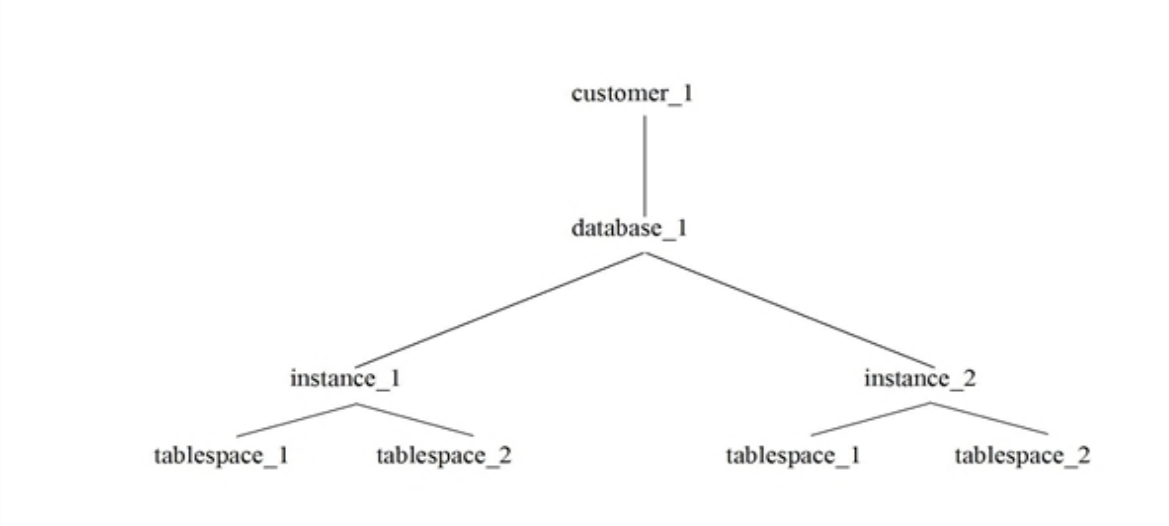
Naming Schema for Services

Service names are unique identifiers or strings which you can choose freely when you write the configuration file. You do not have to create a new policy or condition for each service you are monitoring. If you devise a structured naming schema for your services, you can use HPOM's predefined variables to match the service name attached to the message with the service names set for Service Navigator. The advantage of this is in possibility to keep your HPOM policies generic.

For example, your IT company is managing several database installations for different customers. You know that each database installation can have several instances, and that each instance has several tablespaces which you want to monitor. You also know that you want to use the same HPOM policy to monitor the tablespaces. So you need to come up with a naming schema that enables you to create unique service names, and enables you to match these names with your policy.

Your service hierarchy draft would look similar to the one in Figure 15:

Figure 15: Service Hierarchy Draft



When you know the general layout you can think about the names. Think of what makes each name unique and then compose the name with this information. In the example in Figure 15, the customer name and the name of the system where the database is installed would uniquely identify your service.

You can tell HPOM these names without hardcoding them in the GUI. HPOM can find out some of this information by resolving variables. For example, you can use the HPOM variable `$MSG_NODE_NAME` which would return the name of the node to which the policy is assigned.

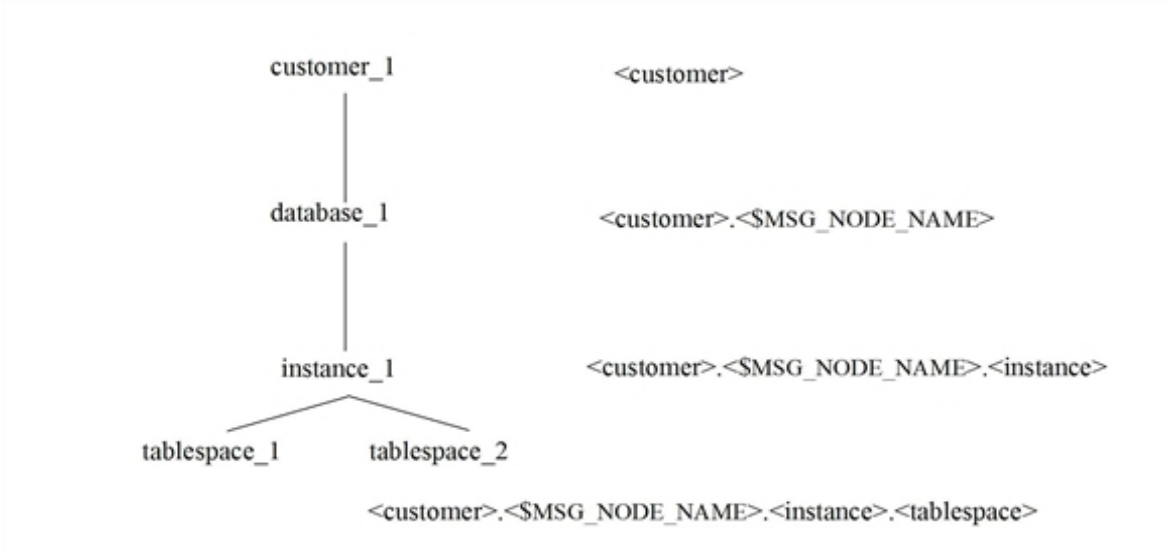
Finding out the customer name depends on the type of policy you are using. If you were using a logfile policy and the name of your customer was included in the logfile, you could use HPOM pattern-matching mechanism to retrieve the customer name. You would use *<*.customer>* with `TEXT` keyword in `CONDITION` block of the policy body and would use *<customer>* with keyword `SERVICE_NAME` keyword in the `SET` block. HPOM then matches `customer` with the name of the customer and would set it as the value of the service name.

HPOM then matches `customer` with the name of customer and would enter it in the `Service Name` field.

In general, you can match any text that your policy or condition provides.

Figure 16 shows the naming schema for your database monitoring services:

Figure 16: Naming Schema



To receive messages for the service `tablespace_1`, use the following with the `SERVICE_NAME` keyword in the policy body:

<customer>.<$MSG_NODE_NAME>.<instance>.<tablespace>

To specify the service name in the service configuration file, type the following:

`Company.system01.your.tech.com.instance_1.tablespace_1`

opcservice Command

The `opcservice` command adds, replaces, and deletes service configurations. It enables you to assign services to operators, or deassign services if they are no longer needed. The location of the

`opcservice` command is as follows:

`/opt/OV/bin/OpC/opcservice`

The `opcservice` command interprets the character set of its command line arguments by the locale setting, so make sure that the locale set for your terminal window corresponds to the system startup language before calling `opcservice`. You can determine the system startup language by reading the following files:

- HP-UX: `/etc/rc.config.d/LANG`
- Sun Solaris: `/etc/default/init`
- Linux: `/etc/sysconfig/i18n`

If the `LANG` environment variable is not set in these files, `C` is used by default.

The output of the `opcservice` command is in the external character set (the character set in which the command was started). The `-xml` option is an exception—if you call `opcservice -xml`, the output is in the external character set UTF-8.

Multiple options can be specified for one command, for example, `opcservice -remove - operators -list - operators` first removes all operators, and then attempts to list them.

For details, see the *opcservice(1m)* manual page.

Tips and Tricks

The information in this section is designed to make your everyday tasks with Service Navigator easier. The information provided covers the following areas:

- "Using Example Configuration Files" on page 397
- "Customizing Icons and Backgrounds" on page 399
- "Customizing Messages in the Message Browser" on page 400

Using Example Configuration Files

Service Navigator comes with a selection of service configuration files which you can copy and edit at your leisure. The example files show how to use the configuration syntax and how to apply Service Navigator to your environment. The examples are available in the following directory on the HP Operations management server:

`/opt/OV/OpC/examples/services/`

This directory also contains a `README` file that lists all available examples and their related files. The following examples are available:

Table 34: Example Configuration Files

File Name	Description
action.xml	Example of how to use actions.
banking.xmla	Banking services example.
carsupply.xmla	Supply chain services example of the car industry.
cltsvr.xml	Client-server system with database and nodes.
cluster.xmla	MC/ServiceGuard cluster system.
diskless.xmla	NFS diskless sample service.
email.xmla	Email sample service using resources.
factor.xml	Factor usage sample service.
fileserv.xml	File server example with file systems and daemons.
georga.xml	Geographically and organizationally structured servers.
icons.xml	Service file that shows all available custom icons.
inet.xml	Network services.
isp.xml	Internet Service Provider service with customers.
local.xml	Example of how to use local service actions.
lvm.xmla	Logical Volume Manager example service.
oracle.xmla	Oracle tablespace monitoring example.
outage.xmla	Outages from Service Navigator as service actions.
perf.xml	Integrating performance applications and messages.
redundant.xmla	Four redundant servers where at least three must be running.
res.xml	Shared and private resources.
sap.xml	The SAP/R3 example from this manual.
vpo.xml[a]	HP Operations Manager as service.

[a]This configuration file can be generated automatically with the shell scripts available in the same directory.

If element values for service files are not specified, warning messages are displayed. To learn how to handle error messages, see HPOM Java GUI Operator's Guide.

The `svcapps.tar` file is a tar archive of an HPOM upload tree that can be uploaded into HPOM. It installs an application group with applications that list, assign, unassign, and remove services and assignments. Add, check, and replace the operations that are not included.

Uploading the Data from the Tar File

To upload the data from the tar file, follow these steps:

1. Change the directory:

   ```
   cd /var/opt/OV/share/tmp/OpC_appl/
   ```

2. Untar the file:

   ```
   tar -xvf /opt/OV/OpC/examples/services/svcapps.tar
   ```

3. Upload the upload data:

   ```
   opccfgupld -add /var/opt/OV/share/tmp/OpC_appl/svcapps
   ```

Customizing Icons and Backgrounds

You can customize the Service Navigator GUI by changing the default background, tree icons, and icons in the service graph. You can do this by specifying your choice in the service configuration file. For more information, see "Service Configuration File Syntax" on page 373.

Some example image files are available after the Service Navigator installation in the following directory on the management server:

`/opt/OV/www/htdocs/ito_op/images/`

All available icons are used in the example service configuration file `/opt/OV/OpC/examples/services/icons.xml`. Activate this file to display each icon in the Service Navigator GUI.

Table 35 lists the areas you can customize.

Table 35: Customizing the Service Navigator GUI

Area	Tag	Input	Size in Pixels	Type	Default
Background graphic	`<Background>`	URL or path on the management server	any, graphics are resized to fit the GUI	GIF	white

Customizing the Service Navigator GUI , continued

Area	Tag	Input	Size in Pixels	Type	Default
Tree icon	`<Icon>`	URL or path on the management server	32 x 32, resized to 16 x 16	GIF[a]	
Graph icon	`<Icon>`	URL or path on the management server	32 x 32, resized to 16 x 16	GIFa	

URLs and graphics for these icons must be well-formed. You can choose between the following transport modes:

- Retrieving files through a web server, as follows:

  `http://<$OPC_MGMTSV>:8081/ITO_OP/images/<icon.gif>`
- Retrieving files from the management server file system, as follows:

  `/opt/OV/www/htdocs/ito_op/images/<icon.gif>`

  You can either specify an absolute or relative path name.

Note: You can use variables as part of the URL. For more information, see "HPOM Variables in URL Definitions" on page 124.

Customizing Messages in the Message Browser

All messages can be identified using service labels and service names. They are displayed in the message browser as columns.

You can also specify service labels and service names as attributes in the Message Properties dialog box, and as columns properties in the message browser configuration file `itoopbrw`.

Additional labels are used to display information that is important for monitoring messages.

Changes in the service label or service name are visible:

- **Immediately**
  - After an operator modifies a message, this message is displayed in the message browser and in the Message Properties dialog box.
  - A new message arrives, after the Administrator specified service label and service name.

[a]Your icons must have a transparent background to properly display the severity status.

If the label is empty because the service is not yet downloaded by Java GUI, you can expand the service in the service tree. This displays the missing service label in the message browser at once.

- **On specified refresh rate**

  HPOM displays messages in the message browser on a specified refresh rate, as set in the Preferences dialog box.

  - After different operators make their changes to a message (for example, taking ownership or changing the service label). The changes are visible on refresh rate.
  - If the attributes in the Message Properties dialog box are updated, they are displayed on refresh rate.

Administrator has to enable the service name and service label on the server side. Otherwise operators cannot customize them on the client side.

If you do not customize the service label and the name (which is disabled by default), or if the service is not loaded because of Service Load on Demand, the columns in the message browser display empty fields.

Example for Customizing Service Labels in the Message Browser Using itooprc

To customize the message browser so that it displays the service labels, set the `show_svc_label_in_msgs` option in the `itooprc` file before the Java GUI startup (the `itooprc` file is located in the user's home directory).

Add, for example, the following line:

```
show_svc_label_in_msgs yes
```

The service labels have a maximum width that is sufficient to display approximately 16 characters. The service names have a maximum width that is sufficient to display approximately 12 characters. If the label or name is too long to be fully displayed, it is cut short and appended with three dots (...).

For more details about customizing service labels, see the *HPOM Java GUI Operator's Guide*.

Customizing Service Submaps

You can customize a service submap by placing service icons on a chosen background (for example, a map of the world) and by positioning the icons to correspond to their real geographical location in a map. A submap can be configured individually for each service.

It is advisable to define backgrounds and icon locations in the service configuration file. See "Setting Service Attributes" on page 371 for information on how you can specify backgrounds and icon positions

in the service configuration file. It is also possible to update these dynamically, as described in "Setting Service Attributes Dynamically" on page 352.

You can configure the following aspects of a service submap:

- **Background image**

  A graphic file with one of the following formats: gif or jpeg. For example, you could choose an image representing a geographical map of the country where your service is located.

  If a background graphic is not specified, the default service submap is displayed. You cannot change the image specification in the service configuration file through the Service Navigator GUI.

- **Background size**

  You can define the size of the background image in pixels, usually in the service configuration file. The size is defined by the image width and height and cannot be changed in the service configuration file through the Service Navigator GUI.

- **Subservice position**

  You can define the position of subservices on the background image of the parent service. The position is defined by specifying values on the X and Y axes, which applies to the display size of the parent service background image.

Any service can have any or all of the above service attributes (Background, Map Size and Position) specified, as any service can simultaneously be a parent service and a subservice.

Table 36: Key Attribute Names for Service Submap Backgrounds

To Configure	Background	Background Size	Subservice Position on Background
Attribute Name	`ov_background`	`ov_map_width` `ov_map_height`	`ov_posX` `ov_posY`
Attribute Value	*`<path_to_image>`* *`<path_to_image>`* is the full path to the image in the file system.	*`<string>`* *`<string>`* must be a positive integer. The value is defined in pixels.	*`<string>`* *`<string>`* must be a positive integer. The value is defined in pixels.

Chapter 10: HPOM Administration UI

In this Chapter

This chapter provides information about architecture, configuration, maintenance, and troubleshooting of the Administration UI.

In this chapter, you can find information about the following topics:

For more detailed information about installation requirements and instructions, see the *HPOM Installation Guide for the Management Server*.

Architecture and References

This section describes the underlying components of the Administration UI.

The information in this section covers the following areas:

Architecture Overview

The Administration UI is implemented based on three-tier architecture.

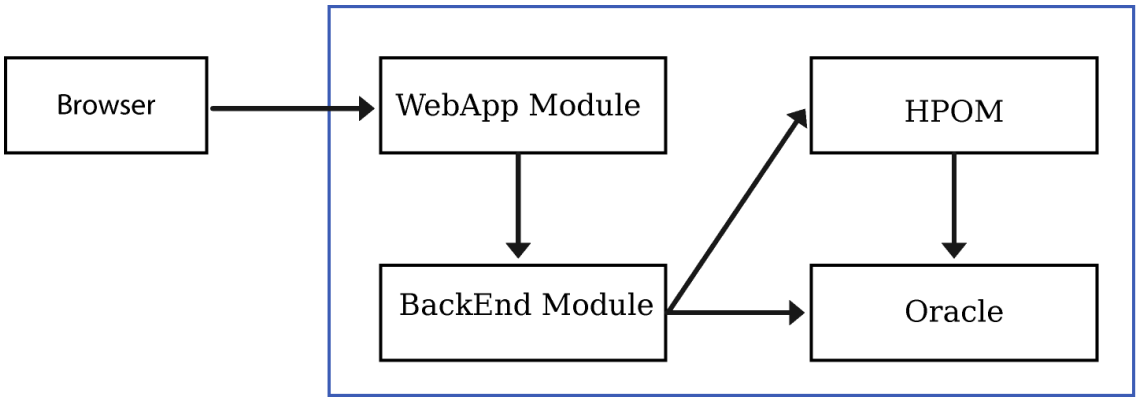
- **Browser**
  The user front-end is a regular web browser such as Mozilla Firefox or Internet Explorer. Therefore, no additional software is required on the end user system. All users can work concurrently.
- **Web Application Module**
  The user connects to the Web Application module (WebApp) through the web browser. The WebApp

module is responsible for generating the dynamic web pages, central data storage, and data processing.

- **BackEnd Module**
  This component interacts with the IT management application (for example, the HP Operations management server). For simple read-only listings (for example, list all policies), the BackEnd module connects directly to the database. For add/modify/delete operations on any HPOM object, the BackEnd module accesses the HPOM API.
  This component also provides the data as XML through a URL schema to the WebApp module.

  The WebApp and BackEnd modules must be installed on the same machine as the management framework server.

Figure 17: :Basic Architecture



Communication and Ports

Communication within the Administration UI is TCP/IP-based. All port numbers listed in this section are the default ports. They are generally defined during the installation of the Administration UI software. However, it is also possible to modify these settings after the installation. For detailed information about advanced scripts for modifying these ports, see "Advanced Tasks" on page 434.

- 9662 (HTTP), 9663 (HTTPS) - The WebApp listens on port 9662 for HTTP requests and on 9663 for HTTPS requests.
  If a firewall exists between the end user network and the HP Operations management server, ports 9662 and 9663 must be opened.

  Some internal services also connect locally to this port (for example, the existing database used to store the Administration UI users, user groups, all user roles). The `solr` service responsible to update the search index also uses port 9662.
- 9661 - All shell commands using `/opt/OV/OMU/adminUI/adminui <cmd>` connect locally to port 9661. Other WebApps or BackEnds or some external CLIs or APIs also use port 9661.

- 9660 - This port is accessed only locally by using the CLI for troubleshooting purposes. No firewall opening is usually needed.
- 32000 - Local communication between the wrapper process (port 31000) and the JRE running in a Java process (for example, `stop`, `dump`, and so on).

Therefore, no port openings are required for ports 9660, 9661, 31000, and 32000 because they are used only locally.

This can be verified by using the `netstat` command, as shown in the following example:

```
[root@deli:/opt/OV/OMU/adminUI/] netstat -an | grep 966
tcp 0 0 *.9660 *.* LISTEN
tcp 0 0 *.9661 *.* LISTEN
tcp 0 0 *.9662 *.* LISTEN
tcp 0 0 *.9663 *.* LISTEN
[root@deli:/opt/OV/OMU/adminUI/] netstat -an | grep 320
tcp 0 0 127.0.0.1.31000 127.0.0.1.32000
ESTABLISHED
tcp 0 0 127.0.0.1.32000 127.0.0.1.31000
ESTABLISHED
tcp 0 0 127.0.0.1.32000 *.* LISTEN
```

Directory Layout Overview

All Administration UI components can be found in the `/opt/OV/OMU/adminUI/` directory in which each component has its own subdirectory.

The most important files and directories are the following:

- `/opt/OV/OMU/adminUI/adminui`
  Central script to control the Administration UI
- `/opt/OV/OMU/adminUI/conf/`
  Configuration files
- `/opt/OV/OMU/adminUI/data/`
  Data location with downloads, XML database, and so on
- `/opt/OV/OMU/adminUI/logs/`
  Log files

Main Directory

The following is a list of all the files and directories inside the main directory:

File or directory	Description
`adminui`	Central control script
`/assemblies`	Available service assemblies
`/backup`	Target directory for local backups
`/bin`	Scripts and binaries
`/checksums`	Checksums location
`/components`	Available Java Business Integration components
`/conf`	Central configuration file location
`/data`	Contains user data (downloads, archive, XML database).
`/datassemblies`	Default data assemblies used for initial initialization of XML database, tasks, and so on.
`/deploy`	Actively deployed assemblies
`/docs`	Third-party open source licenses
`/install`	Actively deployed JBI components
`installation.log`	Second installation log
`/jre`	Bundled JAVA SDK
`/lib`	Shared jar files and native libraries
`/logs`	Log, audit, and task log files
`midas_env.sh`	Contains environment variables.
`midas_server.pid`	Contains server PID at run time.
`run.xml`	Ant file used by `adminui`
`/webapps`	Deployed WebApp
`/webassemblies`	Deployed `webassemblies`
`/work`	ServiceMix deployment and temporary files
`wrapper`	Service wrapper needed to run the application.

Configuration Directory

The default location for all configuration data is `/opt/OV/OMU/adminUI/conf`. The following is a list of all the files and directories inside the configuration directory:

File or directory	Description
`ant/config.xml`	Configuration file for ant tasks
`auth.properties`	SU layer configuration files
`auth.xml`	SU layer configuration files
`backend_local.xml`	Main local backend configuration and capabilities
`becore.properties`	Product feature configuration file
`cocoon.properties`	SU layer configuration files
`config.properties`	Main local configuration file
`core.properties`	SU layer configuration files
`data_local.xml`	Belongs to `./datassemblies`
`derby.properties`	
`exec.properties`	SU layer configuration files
`file.properties`	SU layer configuration files
`fonts/`	
`groovy/`	Do not edit.
`jetty.properties`	SU layer configuration files
`jetty.xml`	SU layer configuration files; `jetty` configuration file for web server configuration (ports, and so on)
`ldap.properties`	Configuration file for LDAP
`local.properties`	SU layer configuration files for the BackEnd adapter
`lock.properties`	SU layer configuration files
`log4j.xml`	Central logging configuration file

File or directory	Description
magic.mime	For mime type detection
midas_analyzer.xml	Configuration file for the adminui analyze command
mime-types.properties	For mime type detection
mime.types	For mime type detection
opccfg.properties	SU layer configuration files
ovcert.properties	SU layer configuration files
ovcoda.properties	SU layer configuration files
ovconfig.properties	SU layer configuration files
ovo/	Do not edit.
ovoappl.properties	SU layer configuration files
ovoconfig.properties	SU layer configuration files, HPOM and database connection settings
ovodistrib.properties	SU layer configuration files
ovoinstall.properties	SU layer configuration files
ovosvc.properties	SU layer configuration files
quartz.properties	Global configuration file for all schedulers
repository/	Contains internal configuration files Do not edit.
schema/	XML schemas for documents
servicemix/	ServiceMix components configuration files including HTTPS
servingxml/	servicingxml configuration files Do not edit.
ssh.properties	SU layer configuration files
stylesheets/	Server side xsl files (for example, adminui backend output)
task.properties	SU layer configuration files

File or directory	Description
terminal.properties	SU layer configuration files
user.properties	SU layer configuration files
usermgmt.properties	SU layer configuration files
velocity.properties	Do not edit.
wacore.properties	Product feature configuration file
webapp.properties	SU layer configuration files

Data Directory

The following list shows the contents of the /opt/OV/OMU/adminUI/data directory that contains user-specific data. If the data directory exists, it is not modified during the installation or uninstallation. If the data directory does not exist, it is created during the installation process.

File or directory	Description
archive/	All archived items go into this directory (as zip or tar files).
clipboard/	All downloads go into this directory.
init/	Data to reset the XML database (loaded when adminui init is run).
path/	Path alias data files Do not edit.
sandbox/	Currently not used.
scratchpad/	Currently not used.
task/	Task data files Do not edit.
txlog/	Work directory for transaction logs of ServiceMix

Log Directory

All main log files are located inside /opt/OV/OMU/adminUI/logs. In general, each component has its own dedicated log file.

The following is a list of all the files and directories inside the log directory:

File or directory	Description
access.log	Access log (details about the client IP and the pages that were accessed)
agent/	Agent installation logs
ant.log	Log for internal ant tasks
audit/	Directory containing daily auditing log files (no rollback, no cleanup) With log level INFO, one line per transaction is logged. With log level DEBUG, everything is logged.
backend.log	Registers activity on the connector on port 9661—it is only written to if DEBUG is enabled.
dead.log	For illegal requests
debug.log	Not used.
events/	Not used.
exist.log	For existing user database
file.log	Not used.
license.log	Not used.
lock.log	Log showing if a lock occurred on a configuration item (WebApp module).
memory.log	Memory consumption
midas.log	Default log if no other special log exists.
nnm.log	Not used.
ovo.log	Logging for HPOM and database (opc_op)
ovoadmin.log	Not used.
package.log	Not used.
performance.log	For support purposes
request.log	For support purposes
requests/	Directory with request logs
results/	Not used.

File or directory	Description
`search.log`	Not used.
`servicemix.log`	For example, if an adapter does not start, check this log first.
`sync.log`	Not used.
`task`	Directory containing individual log files written during the execution of tasks (commands, downloads).
`task.log`	General log used to register if a task was run.
`threadinfo.log`	For support purposes.
`usermgmt.log`	User log when AD or LDAP is used.
`vcs.log`	Not used.
`velocity.log`	Used internally.
`web.log`	WebApp log
`wrapper.log`	Log for the wrapper module
`xmldb.log`	XML database log

The most important log files regarding troubleshooting are the following:

- wrapper.log
- servicemix.log
- midas.log

The `wrapper.log` and `servicemix.log` files belong to the two core components required to successfully run the application.

For more information about error analysis and troubleshooting, see "Troubleshooting" on page 490.

Default Passwords

The default users that already exist in the Administration UI can be divided into the following two groups:

- Users with varying user rights who are able to access HPOM.
- Users for internal administration purposes only (for example, XML database access). These users cannot access HPOM.

The following is a list of all the users and their default passwords:

Module	User	Password
Web Application preconfigured users	admin	secret
	ompolicy_adm	secret
	opc_adm	OpC_adm
	readonly	secret
BackEnd	opc_op	*Defined during the installation.*

Maintaining Administration UI

This section describes all basic shell commands for operating the application:

- "Command Overview" on page 412
- "Administration UI Commands in Detail" on page 414

For specific advanced tasks that are not used on a daily basis, but otherwise require a lot of manual configuration work, special scripts are available. For example, these advanced scripts are available for port, hostname, or password changes. For details, see "Advanced Tasks" on page 434.

This chapter also contains the information about the Java GUI integration and auditing. For details, see the following sections:

- "HPOM Integration " on page 441
- "Auditing" on page 456

For troubleshooting information, see "Troubleshooting" on page 490.

Command Overview

To obtain a list of all available command options, run the `adminui` command without any option:

```
/opt/OV/OMU/adminUI/adminui
```

The following is a list of all available command options:

Option	Description
ant	Runs an Ant task with the built-in Ant.
analyze	Shows configuration and analyzes log files for common errors.
backend	Shows details of local backend.
backup	Backs up configuration (correspondent option: `restore`).

Option	Description
`checksum`	Generates a checksum. Used internally.
`clean`	Removes log files and work files (for example, in case of corruption).
`config`	Shows the configuration of the components.
`deinstall`	Removes a patch (corresponding option: `patch`).
`download`	Downloads and saves user management configuration data out of XML database (corresponding option: `upload`).
`groovy`	Runs a groovy script. Used internally.
`help`	Shows the usage.
`init [force]`	Initializes the XML database. Keep in mind that it deletes the old configuration.
`import`	Imports the file into the clipboard.
`machtypes`	Updates machine type data.
`password`	Password tool
`patch`	Applies a patch (corresponding option: `deinstall`).
`ping`	Pings the server backend.
`reconfigure`	Reconfigures Administration UI properties.
`reload`	Reloads configuration from save (can be from other BackEnd).
`restart`	Restarts the server.
`restore`	Restores configuration from backup (from the same BackEnd).
`servicemix`	Shows ServiceMix deployments.
`start`	Starts the server (options: `-nodeamon -clean`).
`status`	Shows the status of the server.
`save`	Saves configuration (correspondent option: `reload`).
`stop`	Stops the server.

Option	Description
`support`	Collects support information.
`upload`	Uploads user management configuration data into the XML database (corresponding option: `download`).
`version`	Shows application version information.
`webassemblies`	Reinstalls all `webassemblies` on a WebApp. This also rebuilds the `midas.war` file and restarts the Administration UI.
`webassemblies.fast`	Reinstalls all `webassemblies`. The `midas.war` file is not rebuilt and there is no Administration UI restart.

Administration UI Commands in Detail

This section describes the following commands:

- `adminui analyze`

  See "Self-Check" on page 415.
- `adminui backend`

  See "Displaying Server Information" on page 416.
- `adminui backup|restore`

  See "Creating and Restoring Backups" on page 418.
- `adminui clean`

  See "Cleanup and File Corruption Fixing" on page 421.
- `adminui config`

  See "Displaying Configuration" on page 422.
- `adminui download|upload`

  See "Downloading User Management Configuration" on page 423.
- `adminui patch|deinstall`

  See "Installing and Removing Patches" on page 424.
- `adminui start|stop|restart`

  See "Starting, Stopping, and Restarting the Administration UI" on page 425.
- `adminui save|reload`

  See "Saving Configuration" on page 425.
- `adminui reconfigure`

See "Reconfiguring Properties" on page 428.

- `adminui status`

  See "Displaying the Server Status" on page 431.

- `adminui support`

  See "Collecting Support Information" on page 432.

- `adminui version`

  See "Displaying the Product Version" on page 434.

Note: The commands that are only for internal use are not described in this section.

Self-Check

The `adminui analyze` command lists the configuration of the Administration UI and checks all log files for common errors. If an error is detected, a troubleshooting tip is displayed. It is useful to run this command and check the result before contacting Product Support.

```
# ./adminui analyze
Configuration values
====================
   Installation:
      Version                          09.22.160 (build: 4070)
      OEM Version                      cvpl
      Installation Directory           /opt/OV/OMU/adminUI
      Installation Type                full
   Server:
      Hostname                         iwfvm07062.hpeswlab.net
      Platform                         unix
      Backend Identifier               iwfvm07062.hpeswlab.net_server
   Communication:
      JMX Port                         9660
      Server Port                      9661
      HTTP Port                        9662
      HTTPS Port                       9663
   XMLDB:
```

```
      XML DB User
   Database postgresql:
      Database Home
      Database Major Version           9
      Database Host                    localhost
      Database Port                    5432
      Database SID                     openview
      Database User                    opc_op
   Operations Manager:
      Version                          922
      Codeset                          UTF-8
   Licenses:
     License Type                     licensed through HPOM


Errors in Logfiles
==================
```

In this example, no errors were found.

Displaying Server Information

The `adminui backend` command is used to display the configuration settings of the local server. For displaying extended configuration information, use the `adminui analyze` or `adminui config` command.

```
# ./adminui backend
...
backend:
   [xslt] Processing /opt/OV/OMU/adminUI/conf/backend_local.xml to /opt/OV/OMU
/adminUI/work/server_20161110004537.txt
   [xslt] Loading stylesheet /opt/OV/OMU/adminUI/conf/stylesheets/server_view.
                                                                        xsl
   [echo]
```

```
   [echo] Server iwfvm07062.hpeswlab.net_server:
   [echo]   Server Identifier: iwfvm07062.hpeswlab.net_server
   [echo]   Hostname: iwfvm07062.hpeswlab.net
   [echo]   Protocol: http
   [echo]   Port: 9661
   [echo]   Secure Communication: false
   [echo]   Platform: unix
   [echo]   Install Directory: /opt/OV/OMU/adminUI
   [echo]   Services:
   [echo]     task
   [echo]     exec
   [echo]     terminal
   [echo]     file
   [echo]     backend
   [echo]     usermgmt
   [echo]     auth
   [echo]     lock
   [echo]     ovoconfig
   [echo]     ovcert
   [echo]     opccfg
   [echo]     ovconfig
   [echo]     ovoappl
   [echo]     ovoinstall
   [echo]     ovosvc
   [echo]     ovcoda
   [echo]     net
   [echo]     notice
   [echo]
  [delete] Deleting: /opt/OV/OMU/adminUI/work/server_20161110004537.txt

BUILD SUCCESSFUL
```

```
Total time: 1 second
```

Creating and Restoring Backups

The `adminui backup` and `adminui restore` commands are used to back up and restore the complete configuration on a local server. During the backup operation, the data is copied into a `.zip` file that is stored in `/opt/OV/OMU/adminUI/`.

Caution: During both activities, the Administration UI must be running. Otherwise, the XML database containing all the user information cannot be accessed.

The backup you want to restore must match the installed Administration UI version you are running. The hostname and the Administration UI identifier should also match. Otherwise, the advanced rename scripts for the hostname and the identifier must be run after the restore.

This means that you cannot restore a backup created under 4.0.0 to your existing 4.1.0 system. A 4.1.1 backup cannot be restored to a 4.0.x or 4.1.0 system. Furthermore, a backup from server A should not be restored to server B if the hostname does not match.

Note: To save and transfer configuration data between systems with different hostnames, it is recommended to use the `adminui save` and `adminui reload` commands. For details, see "Saving Configuration" on page 425.

The backup includes:

- XML database with Administration UI users, groups, and roles
- Path aliases
- Tasks
- All configuration files in `/opt/OV/OMU/adminUI/conf`

Creating a Backup

```
# ./adminui backup work/backup_20161011922
[...]
backup:
test.app.running:
     [echo] Verify that all Administrator UI services are up.
```

```
     [echo] Trying to reach
http://iwfvm07062.hpeswlab.net:9662/midas/checkServices

backup:

    [mkdir] Created dir: /opt/OV/OMU/adminUI/work/backup_20161110042950

     [echo] backing up HPOM Administration UI configuration to
/opt/OV/OMU/adminUI/work/backup_20161110042950

intern.backup_xmldb:

     [echo] saving XML DB

    [mkdir] Created dir: /opt/OV/OMU/adminUI/work/backup_20161110042950/xmldb

     [copy] Copying 3 files to /opt/OV/OMU/adminUI/work/backup_
20161110042950/xmldb

intern.backup_cvs:

intern.backup_conf:

     [echo] saving configuration from /opt/OV/OMU/adminUI/conf

    [mkdir] Created dir: /opt/OV/OMU/adminUI/work/backup_20161110042950/conf

     [copy] Copying 188 files to /opt/OV/OMU/adminUI/work/backup_
20161110042950/conf

     [copy] Copied 7 empty directories to 1 empty directory under
/opt/OV/OMU/adminUI/work/backup_20161110042950/conf

intern.backup_path:

     [echo] saving path aliases from /opt/OV/OMU/adminUI/data/path

    [mkdir] Created dir: /opt/OV/OMU/adminUI/work/backup_
20161110042950/data/path

     [copy] Copying 74 files to /opt/OV/OMU/adminUI/work/backup_
20161110042950/data/path

intern.backup_task:

     [echo] saving tasks from /opt/OV/OMU/adminUI/data/task

    [mkdir] Created dir: /opt/OV/OMU/adminUI/work/backup_
20161110042950/data/task

     [copy] Copying 99 files to /opt/OV/OMU/adminUI/work/backup_
20161110042950/data/task

intern.backup_zip:

      [zip] Building zip: /opt/OV/OMU/adminUI/work/backup_20161011922/backup_
20161110042950.zip

    [delete] Deleting directory /opt/OV/OMU/adminUI/work/backup_20161110042950
```

```
     [echo] backup archived in work/backup_20161011922/backup_20161110042950.zip
BUILD SUCCESSFUL
Total time: 2 seconds
```

In the last lines, the name and the location of the backup ZIP file are displayed.

Restoring a Backup

To restore a backup, use the `adminui restore` command. You must also provide the path of the backup ZIP file created with an earlier backup.

Caution: Because the server is restarted during the restore, make sure to inform the users about the downtime.

The following example shows that an automatic stop-clean-start is performed when data restore is complete, so the restored data is used:

```
# ./adminui restore ./backup_20161110042950.zip
[...]
restore:
    [mkdir] Created dir: /opt/OV/OMU/adminUI/work/restore_20161110043130
     [echo] restoring into file /opt/OV/OMU/adminUI/work/restore_20161110043130
     [echo] restoring backup work/backup_20161011922/backup_20161110042950.zip
    [unzip] Expanding: /opt/OV/OMU/adminUI/work/backup_20161011922/backup_
20161110042950.zip into /opt/OV/OMU/adminUI/work/restore_20161110043130
intern.restore_xmldb:
     [echo] restoring XML DB
     [echo] from path /opt/OV/OMU/adminUI/work/restore_20161110043130
     [echo] to Path /opt/OV/OMU/adminUI/data/xmldb
     [copy] Copying 3 files to /opt/OV/OMU/adminUI/data/xmldb
intern.restore_conf:
     [echo] restoring configuration to /opt/OV/OMU/adminUI/conf
     [copy] Copying 188 files to /opt/OV/OMU/adminUI/conf
intern.restore_path:
     [echo] restoring path aliases to /opt/OV/OMU/adminUI/data/path
```

```
     [copy] Copying 74 files to /opt/OV/OMU/adminUI/data/path
intern.restore_task:
     [echo] restoring tasks to /opt/OV/OMU/adminUI/data/task
     [copy] Copying 99 files to /opt/OV/OMU/adminUI/data/task
[...]
     [echo] restarting server
intern.server_stop.unix:
     [echo] Stopping server
     [exec] clean:
intern.server_start.unix:
     [echo] Starting server
intern.server_start.windows:
     [echo] restore successfull
BUILD SUCCESSFUL
Total time: 11 seconds
```

Cleanup and File Corruption Fixing

The `adminui clean` command removes all log files in the `/opt/OV/OMU/adminUI/logs` and `/opt/OV/OMU/adminUI/work` directories. The function of the `work` directory is similar to a cache. At application startup, all service assemblies are unpacked into this directory. Because file corruption can occur inside the `work` directory at run time, it is recommended to run the `adminui analyze` command followed by clean in case of starting problems.

After you run the `adminui clean` command, the Administration UI application is not started automatically. You must start it manually by running the `adminui start` command.

```
# ./adminui clean
[...]
Buildfile: /opt/OV/OMU/adminUI/run.xml
clean:
BUILD SUCCESSFUL
Total time: 1 second
```

Displaying Configuration

The `adminui config` command shows the information about server settings and configuration of all installed adapters and components. It can be used to find out the settings specified during the installation such as ports or hostnames of all deployed components.

```
# ./adminui config
[...]
Configuration values
====================
   Installation:
     Version                          09.22.160 (build: 4070)
     OEM Version                      cvpl
     Installation Directory           /opt/OV/OMU/adminUI
     Installation Type                full
   Server:
     Hostname                         iwfvm07062.hpeswlab.net
     Platform                         unix
     Backend Identifier               iwfvm07062.hpeswlab.net_server
   Communication:
     JMX Port                         9660
     Server Port                      9661
     HTTP Port                        9662
     HTTPS Port                       9663
   XMLDB:
     XML DB User
   Database postgresql:
     Database Home
     Database Major Version           9
     Database Host                    localhost
     Database Port                    5432
     Database SID                     openview
```

```
    Database User                   opc_op
  Operations Manager:
    Version                         922
    Codeset                         UTF-8
  Licenses:
    License Type                    licensed through HPOM
```

Downloading User Management Configuration

The `adminui download` command enables you to download the Administration UI user management configuration data as defined by an index file. In the index file, you define which data is downloaded or uploaded.

The syntax is as follows:

- Download the user data dependent on index configuration:

  ```
  cd /opt/OV/OMU/adminUI/
  ./adminui download <indexfile> <targetdirectory>
  ```

- Upload user data dependent on the directory:

  ```
  cd /opt/OV/OMU/adminUI/
  ./adminui upload [add|modify]<directory>
  ```

Subentities are not implemented, but this should affect only user roles.

The index file must be in the following format:

```
<index product=”Administration UI” version=”9.2.2”>
<!-- users to download -->
<um:userref>
<um:name>opc_adm</um:name>
</um:userref>
<!-- groups to download -->
<um:usergroupref>
<um:name>administrators</um:name>
</um:usergroupref>
<!-- roles to download -->
<um:roleref>
<um:name>administrator_role</um:name>
</um:roleref>
</index>
```

Installing and Removing Patches

The `adminui patch` command is used for installing Administration UI patches that you receive in a `.zip` format (that is, `<patch_ID>.zip`). For example, `OMUADMINUI_00003.zip`.

To install such a patch file, copy the ZIP file into the installation directory `/opt/OV/OMU/adminUI/` or any other location (for example, `/tmp`). Make sure that you do not unzip it because this is done by the patch mechanism itself.

To install a patch, run the following command:

```
/opt/OV/OMU/adminUI/adminui patch /tmp/<patch_ID>.zip
```

If the patch ZIP file is located inside `/opt/OV/OMU/adminUI/`, run the following command:

```
/opt/OV/OMU/adminUI/adminui patch <patch_ID>.zip
```

For example:

```
/opt/OV/OMU/adminUI/adminui patch OMUADMINUI_00003.zip
```

To offer a rollback mechanism, the existing configuration files are backed up in a new directory inside `/opt/OV/OMU/adminUI/<patch_ID>`. For example, `/opt/OV/OMU/adminUI/OMUADMINUI_00003` before the patch is applied.

The `adminui deinstall` command removes a previously installed patch. You must specify the ID of the patch to be deinstalled. Run the following command:

```
/opt/OV/OMU/adminUI/adminui deinstall <patch_ID>
```

Caution: When removing a patch, make sure to enter only the patch ID (that is, without the `.zip` extension). For example:

```
/opt/OV/OMU/adminUI/adminui deinstall OMUADMINUI_00003
```

The ID is defined by the directory name in which the backup files are located.

Note: Run the `adminui backup` command to create a backup for the newly patched latest version because backups for older versions cannot be used for a restore.

For details, see "Creating and Restoring Backups" on page 418.

Restart the Administration UI software manually by running the following commands:

```
/opt/OV/OMU/adminUI/adminui clean
```

```
/opt/OV/OMU/adminUI/adminui start
```

Caution: Because the server must be restarted after installing or removing a patch, make sure to inform the users about the downtime.

Starting, Stopping, and Restarting the Administration UI

Depending on whether you want to start, stop, or restart the Administration UI on the local system, run one of the following commands:

```
/opt/OV/OMU/adminUI/adminui stop
/opt/OV/OMU/adminUI/adminui start
/opt/OV/OMU/adminUI/adminui restart
```

Caution: The `adminui start` command returns to the shell prompt immediately. The Administration UI is, however, still starting up. Depending on the speed of the local system, the startup may take up to a few minutes.

Note: To check the start-up progress, type:

```
tail -f /opt/OV/OMU/adminUI/logs/wrapper.log
tail -f /opt/OV/OMU/adminUI/logs/servicemix.log
```

Generally, if no more logging takes place inside `wrapper.log` and `servicemix.log`, the application startup is complete.

Saving Configuration

The `adminui save|reload` command is a stripped-down version of the adminui backup|reload command. In contrast to the `adminui backup|restore` command, configuration data from `/opt/OV/OMU/adminUI/conf` is not saved.

The backup from `/opt/OV/OMU/adminUI/adminui` save includes:

- XML database with users, user groups, and user roles
- Path alias
- Tasks

Therefore, you can use this command to save the above-mentioned configuration on server A and reload the data on server B.

The backup is placed inside `/opt/OV/OMU/adminUI/save_<datestamp>.zip`

Caution: During both activities, the Administration UI must be running. Otherwise, the XML database cannot be accessed.

```
# ./adminui save
```

```
[...]

save:

    [mkdir] Created dir: /opt/OV/OMU/adminUI/work/backup_20161110050050

     [echo] saving HPOM Administration UI configuration to
/opt/OV/OMU/adminUI/work/backup_20161110050050

intern.save_xmldb:

     [echo] saving XML DB

    [mkdir] Created dir: /opt/OV/OMU/adminUI/work/backup_20161110050050/xmldb

     [copy] Copying 3 files to /opt/OV/OMU/adminUI/work/backup_
20161110050050/xmldb

intern.backup_path:

     [echo] saving path aliases from /opt/OV/OMU/adminUI/data/path

    [mkdir] Created dir: /opt/OV/OMU/adminUI/work/backup_
20161110050050/data/path

     [copy] Copying 74 files to /opt/OV/OMU/adminUI/work/backup_
20161110050050/data/path

intern.backup_task:

     [echo] saving tasks from /opt/OV/OMU/adminUI/data/task

    [mkdir] Created dir: /opt/OV/OMU/adminUI/work/backup_
20161110050050/data/task

     [copy] Copying 99 files to /opt/OV/OMU/adminUI/work/backup_
20161110050050/data/task

intern.backup_zip:

    [zip] Building zip: /opt/OV/OMU/adminUI/backup/save_20161110050050.zip

     [delete] Deleting directory /opt/OV/OMU/adminUI/work/backup_20161110050050

     [echo] save archived in /opt/OV/OMU/adminUI/backup/save_20161110050050.zip

BUILD SUCCESSFUL

Total time: 4 seconds
```

In the last lines, you find the file name and its location.

Reloading the Data

To reload the data back into the Administration UI, use the `adminui reload` command as follows:

`/opt/OV/OMU/adminUI/adminui reload save_<timestamp>.zip`

Caution: Because the server is restarted during the reload, make sure to inform the users about the downtime.

As you can see in the following output example, the `adminui reload` command restarts the Administration UI at the end of the operation:

```
#./adminui reload save_20161110050050.zip

[...]

reload:

    [mkdir] Created dir: /opt/OV/OMU/adminUI/work/restore_20161110051046

     [echo] reloading saved configuration from backup/save_20161110050050.zip

    [unzip] Expanding: /opt/OV/OMU/adminUI/backup/save_20161110050050.zip into
/opt/OV/OMU/adminUI/work/restore_20161110051046

intern.reload_xmldb:

     [echo] restoring XML DB

     [copy] Copying 3 files to /opt/OV/OMU/adminUI/data/xmldb

intern.restore_path:

     [echo] restoring path aliases to /opt/OV/OMU/adminUI/data/path

     [copy] Copying 74 files to /opt/OV/OMU/adminUI/data/path

intern.restore_task:

     [echo] restoring tasks to /opt/OV/OMU/adminUI/data/task

     [copy] Copying 99 files to /opt/OV/OMU/adminUI/data/task

intern.backend_update:

     [echo] Updating backend endpoints

    [mkdir] Created dir: /opt/OV/OMU/adminUI/work/20161110051046

     [copy] Copying 1 file to /opt/OV/OMU/adminUI/work/20161110051046

    [zip] Building zip: /opt/OV/OMU/adminUI/work/20161110051046/midas-wahttp-
sa.zip

     [copy] Copying 1 file to /opt/OV/OMU/adminUI/assemblies

     [copy] Copying 1 file to /opt/OV/OMU/adminUI/deploy

     [echo] Updated backend endpoints

   [delete] Deleting directory /opt/OV/OMU/adminUI/work/20161110051046

     [echo] restarting server

intern.server_stop.unix:
```

```
     [echo] Stopping server
intern.server_start.unix:
     [echo] Starting server
intern.server_start.windows:
     [echo] reload successfull
BUILD SUCCESSFUL
Total time: 11 seconds
```

Reconfiguring Properties

The `adminui reconfigure` command reads the current DBMajorVersion, DB host, DB port, ovoVersion and allows user to edit them.

The following is an example of reconfiguring the Administration UI properties.

```
#./adminui reconfigure
Reconfiguration of the following Administration UI configuration properties is
supported:-
-Database Major Version (DBMajorVersion)
-Database Hostname (MIDAS_DB_HOST)
-Database Port (MIDAS_DB_PORT)
-AdminUI OVO Version (ovoVersion)
User discretion advised, setting incorrect values may hinder Administration UI
start-up.
Continue? ["no",yes]
  > yes
-Current DBMajorVersion is '11', enter desired DBMajorVersion value ["11"]
  > 12
-Current Database Hostname is 'localhostname.domain.com', enter desired Database
Hostname value ["localhostname.domain.com"]
  > remotedatabase.domain.com
-Current Database port is '1521', enter desired Database port value ["1521"]
  > 1522
-Current ovoVersion value is '922', enter desired ovoVersion value ["922"]
```

```
  > 922


Summary

The following changes will be made:

-DBMajorVersion will be changed from 11 to 12

-Database Hostname will be changed from localhostname.domain.com to
remotedatabase.domain.com

-Database Port will be changed from 1521 to 1522

-ovoVersion will be changed from 922 to 922

Continue? ["no", yes]

  > yes


Changing DBMajorVersion to 12 in the following files:

/opt/OV/OMU/adminUI/conf/opccfg.properties

/opt/OV/OMU/adminUI/conf/ovoappl.properties

/opt/OV/OMU/adminUI/conf/ovoconfig.properties


Backup of modified files:

/opt/OV/OMU/adminUI/conf/opccfg.properties-BACKUP20161111125926

/opt/OV/OMU/adminUI/conf/ovoappl.properties-BACKUP20161111125926

/opt/OV/OMU/adminUI/conf/ovoconfig.properties-BACKUP20161111125926


Files after editing:

/opt/OV/OMU/adminUI/conf/opccfg.properties:ovodb.DBMajorVersion=12

/opt/OV/OMU/adminUI/conf/ovoappl.properties:ovodb.DBMajorVersion=12

/opt/OV/OMU/adminUI/conf/ovoconfig.properties:ovodb.DBMajorVersion=12


Changing Database Hostname to remotedatabase.domain.com in the following files:

/opt/OV/OMU/adminUI/conf/opccfg.properties

/opt/OV/OMU/adminUI/conf/ovoappl.properties

/opt/OV/OMU/adminUI/conf/ovoconfig.properties

/opt/OV/OMU/adminUI/conf/ovoinstall.properties
```

```
Backup of modified files:
/opt/OV/OMU/adminUI/conf/opccfg.properties-BACKUP20161111125927
/opt/OV/OMU/adminUI/conf/ovoappl.properties-BACKUP20161111125927
/opt/OV/OMU/adminUI/conf/ovoconfig.properties-BACKUP20161111125927
/opt/OV/OMU/adminUI/conf/ovoinstall.properties-BACKUP20161111125927

Files after editing:
/opt/OV/OMU/adminUI/conf/opccfg.properties:ovodb.url=jdbc\:oracle\:thin\:@remote
database.domain.com\:1521\:openview
/opt/OV/OMU/adminUI/conf/ovoappl.properties:ovodb.url=jdbc\:oracle\:thin\:@remot
edatabase.domain.com\:1521\:openview
/opt/OV/OMU/adminUI/conf/ovoconfig.properties:ovodb.url=jdbc\:oracle\:thin\:@rem
otedatabase.domain.com\:1521\:openview
/opt/OV/OMU/adminUI/conf/ovoinstall.properties:ovodb.url=jdbc\:oracle\:thin\:@re
motedatabase.domain.com\:1521\:openview

Changing Database Port to 1522 in the following files:
/opt/OV/OMU/adminUI/conf/opccfg.properties
/opt/OV/OMU/adminUI/conf/ovoappl.properties
/opt/OV/OMU/adminUI/conf/ovoconfig.properties
/opt/OV/OMU/adminUI/conf/ovoinstall.properties

Backup of modified files:
/opt/OV/OMU/adminUI/conf/opccfg.properties-BACKUP20161111125928
/opt/OV/OMU/adminUI/conf/ovoappl.properties-BACKUP20161111125928
/opt/OV/OMU/adminUI/conf/ovoconfig.properties-BACKUP20161111125928
/opt/OV/OMU/adminUI/conf/ovoinstall.properties-BACKUP20161111125928

Files after editing:
/opt/OV/OMU/adminUI/conf/opccfg.properties:ovodb.url=jdbc\:oracle\:thin\:@remote
database.domain.com\:1522\:openview
```

```
/opt/OV/OMU/adminUI/conf/ovoappl.properties:ovodb.url=jdbc\:oracle\:thin\:@remot
edatabase.domain.com\:1522\:openview

/opt/OV/OMU/adminUI/conf/ovoconfig.properties:ovodb.url=jdbc\:oracle\:thin\:@rem
otedatabase.domain.com\:1522\:openview

/opt/OV/OMU/adminUI/conf/ovoinstall.properties:ovodb.url=jdbc\:oracle\:thin\:@re
motedatabase.domain.com\:1522\:openview


Administration UI is now configured with the following parameter-values:-

-DBMajorVersion    = 12

-Database Hostname = remotedatabase.domain.com

-Database Port     = 1522

-ovoVersion        = 922


Please restart Administration UI for the changes to take effect.

Backed up files (if any) may be moved or deleted post verification of
Administration UI status.
```

Displaying the Server Status

The `adminui status` command displays the status of the processes including the connection status and whether the server assembly is correct or not.

An extract

```
# ./adminui status
 [...]

intern.status_service:

   [echo] sending status request to service ovoconfig...

   [copy] Copying 1 file to /opt/OV/OMU/adminUI/work/20161110005806

 [reqpost] sending request to http://iwfvm07062.hpeswlab.net:9661/midas_client/

[...]

   [echo] status of backend iwfvm07062.hpeswlab.net_server

   [echo]

   [echo]     Server:  iwfvm07062.hpeswlab.net_server

   [echo]     Service: auth
```

```
   [echo]       Name:        User Authentication Filter
   [echo]       Status:      connected
   [echo]       Error count: 0
   [echo]       Status:      unlicensed
   [echo]
   [echo]
   [echo]     Server:  iwfvm07062.hpeswlab.net_server
   [echo]     Service: backend
   [echo]       Name:        Local Backend Server
   [echo]       Status:      connected
   [echo]       Error count:
   [echo]       Status:      unlicensed
[...]
BUILD SUCCESSFUL
Total time: 30 seconds
```

If the application is not running correctly, a `BUILD FAILED` message is received.

Collecting Support Information

In case of technical problems, HP Support requires the detailed information about the individual configuration of the Administration UI. The `adminui support` command enables you to collect all required log files and configuration files, type:

```
/opt/OV/OMU/adminUI/adminui support
```

The shell output looks as in the following example:

```
# ./adminui support
[...]
support.zip:
     [echo] collecting support information ...
     [echo] collecting version info ...
     [echo] collecting installed files ...
     [echo] collecting Java properties ...
```

```
[propertyfile] Creating new property file: /opt/OV/OMU/adminUI/work/env_
20161110022725.properties
intern.unix_uname:
     [echo] checking uname ...
intern.unix_env:
     [echo] collecting environment ...
intern.win_env:
intern.linux_procinfo:
     [echo] collecting processor info ...
intern.ovo7_info:
intern.ovo8_info:
     [echo] collecting HPOM for UNIX info ...
     [copy] Copying 1 file to /opt/OV/OMU/adminUI/work
     [copy] Copying 1 file to /opt/OV/OMU/adminUI/work
intern.truststore_info:
     [echo] collecting trustore info
     [exec] Result: 1
intern.backend_info:
intern.unix_uname:
     [echo] checking uname ...
intern.ifconfig:
     [echo] checking IP configuration ...
intern.lanscan:
intern.ipconfig:
intern.oracle_config:
     [echo] collecting oracle configuration ...
     [copy] Warning: Could not find file /network/admin/listener.ora to copy.
intern.analyze_unix:
     [echo] running analyze ...
intern.analyze_windows:
     [echo] creating support zip ...
    [zip] Building zip: /opt/OV/OMU/adminUI/support_20161110022725.zip
```

```
     [echo] cleaning up ...

     [echo] send the file /opt/OV/OMU/adminUI/support_20161110022725.zip to HP
support

BUILD SUCCESSFUL

Total time: 3 minutes 17 seconds

At the end of the output, you see the support file name and the location.
```

Displaying the Product Version

The `adminui version` command is used for displaying the version and the build. Type:

`/opt/OV/OMU/adminUI/adminui version`

```
# ./adminui version

version:

     [echo] installed product = HP Operations Manager Administration UI (HPOM
Administration UI)

     [echo] HPOM Administration UI version = 09.22.160

     [echo] HPOM Administration UI build number = 4070

     [echo] HPOM Administration UI build date = 20161031

     [echo] HPOM Administration UI install date = 11/2/16 7:21 AM

     [echo] HPOM Administration UI installation directory = /opt/OV/OMU/adminUI

     [echo] installed HPOM Administration UI products:

     [echo]   HPOM Administration UI Documentor Backend Server

     [echo]   HPOM Administration UI Configurator Backend Server

     [echo]   HPOM Administration UI Light Web Application Server

     [echo]   HPOM Administration UI Web Application

BUILD SUCCESSFUL

Total time: 1 second
```

Advanced Tasks

For specific advanced tasks that are not used on a daily basis, but otherwise require a lot of manual configuration work, special scripts are available (for example, for port, hostname, or password changes).

This section describes the following advanced tasks:

Updating Machtypes

When new machtypes are introduced in HPOM with patches or a new agent version, the BackEnd module reads them dynamically. Therefore, no manual command execution is needed there.

Update the WebApp module updated manually by running the following command:

```
/opt/OV/OMU/adminUI/adminui machtypes
```

No restart of the WebApp module is necessary, so there is no downtime.

Updating the opc_op Password

When the `opc_op` password is changed and updated, it is also necessary to do this in the Administration UI. Otherwise, all list operations (for example, list Policy Bank, list All Nodes) fail and an error message appears.

Update the `opc_op` password as follows:

1. Run the following command:

   ```
   /opt/OV/OMU/adminUI/adminui password -u ovodb -a -p <password>
   ```

2. Restart the application by running these commands:

   ```
   /opt/OV/OMU/adminUI/adminui clean
   ```

   ```
   /opt/OV/OMU/adminUI/adminui start
   ```

For details, see "Accessing HPOM and the Database" on page 449 and "Problems with Passwords" on page 445.

Importing HPOM Download Data

The `adminui import` command is used to import an HPOM configuration download directory into the Administration UI Clipboard directory. From this directory, it can be processed further using regular GUI functionality.

For example:

```
/opt/OV/OMU/adminUI/adminui import /tmp/my_download
```

To access the Clipboard directory, from the Administrative menu, select **Browse > Downloads**.

Renaming the BackEnd Identifier

The BackEnd identifier that is stored and maintained in multiple configuration files is a central attribute of each system. The syntax is usually `hostname_server`. If the identifier must be changed (usually in conjunction with a rename of the hostname), run the following command:

```
/opt/OV/OMU/adminUI/adminui ant -f conf/ant/admin.xml rename_backend -
Dbackend=<newname>
```

Caution: Because the server is restarted during the rename, make sure to inform the users about the downtime.

Changing the Hostname

The hostname is stored in multiple configuration files. Therefore, to avoid potential problems, make sure that you run the following command before (recommended) or after the hostname change:

```
/opt/OV/OMU/adminUI/adminui ant -f conf/ant/admin.xml rename_hostname -
Dhost=<newhost>
```

Caution: Because the server is restarted during the rename, make sure to inform the users about the downtime.

If the `rename_hostname` command is run before the hostname change, you can restart the server to complete the procedure of changing the hostname.

However, if the `rename_hostname` command is run after the hostname change, it is possible that the Administration UI will not work properly. In this case it is recommended to restart the Administration UI manually and run the `rename_hostname` command again.

Changing the BackEnd Port (9661)

To change the BackEnd port (default 9661), follow these steps:

1. Run the following command:

   ```
   /opt/OV/OMU/adminUI/adminui ant -f conf/ant/admin.xml change_server_port -
   Dport=<new-port>
   ```

2. Restart the Administration UI by running the following commands:

   ```
   /opt/OV/OMU/adminUI/adminui clean
   ```

   ```
   /opt/OV/OMU/adminUI/adminui start
   ```

Caution: Make sure to inform the users about the downtime.

Changing the WebApp HTTP or HTTPS Port

To change the HTTP (default 9662) or HTTPS (default 9663) port, follow these steps:

1. Run one of the following commands:
   - HTTP port change only:

     ```
     /opt/OV/OMU/adminUI/adminui ant -f conf/ant/admin.xml change_web_port -
     Dport.http=<new-port>
     ```

   - HTTPS port change only:

     ```
     /opt/OV/OMU/adminUI/adminui ant -f conf/ant/admin.xml change_web_port -
     Dport.https=<new-port>
     ```

   - Combined HTTP and HTTPS port change:

     ```
     /opt/OV/OMU/adminUI/adminui ant -f conf/ant/admin.xml change_web_port -
     Dport.http=<new-port> -Dport.https=<new-port2>
     ```

2. Restart the Administration UI by running the following commands:

   ```
   /opt/OV/OMU/adminUI/adminui clean
   ```

   ```
   /opt/OV/OMU/adminUI/adminui start
   ```

Caution: Make sure to inform the users about the downtime.

Checking the WebApp HTTP or HTTPS Port

To check if the HTTP and HTTPS ports are set correctly in the `config.properties` and `jetty.xml` configuration files, run the following command:

```
/opt/OV/OMU/adminUI/adminui ant -f conf/ant/admin.xml check.webapp_ports
```

You can also use the following optional parameters:

- `wait.for`

  Defines the time in seconds to wait for the WebApp to start. The default value is `5`.

- `execute.change_web_port.task`

  Runs an automatic task for changing the web ports in case a problem is detected during the port check.

Examples of the commands:

```
/opt/OV/OMU/adminUI/adminui ant -f conf/ant/admin.xml check.webapp_ports -
Dwait.for=100
```

```
/opt/OV/OMU/adminUI/adminui ant -f conf/ant/admin.xml check.webapp_ports -
Dexecute.change_web_port.task=yes
```

```
/opt/OV/OMU/adminUI/adminui ant -f conf/ant/admin.xml check.webapp_ports -
Dexecute.change_web_port.task=yes -Dwait.for=100
```

Disabling the WebApp HTTP Port (9662)

To access the Administration UI through a web-browser, you can choose between HTTP (the default port is 9662) and HTTPS (the default port is 9663).

To disable HTTP access, you can bind port 9662 to `localhost`. To do so, follow these steps (assuming that the Administration UI is up and running):

1. Edit the following file:

   `/opt/OV/OMU/adminUI/conf/jetty.xml`

2. Inside the `jetty.xml` file, search for the following block:

   ```
   <!-- SSL context factory -->
   <bean autowire="default" class="com.bes.itm.comp.jetty.SslPropertiesConfig"
   dependency-check="default" id="sslPropertiesConfig" lazy-init="default">
   ```

3. Search for the following line in the block:

   ```
   <property name="host" value=""/>
   ```

   a. If the line is present, set `value` as `localhost`.

   b. If the line is not present, add the following line:

   ```
   <property name="host" value="localhost"/>
   ```

   The block looks as follows after the modification:

   ```
   <!-- SSL context factory -->
   <bean autowire="default" class="com.bes.itm.comp.jetty.SslPropertiesConfig"
   dependency-check="default" id="sslPropertiesConfig" lazy-init="default">
     <constructor-arg>
        <ref bean="sslContextFactory"/>
     </constructor-arg>
     <property name="host" value="localhost"/>
     <property name="httpPort" value="9662"/>
   ```

```
  <property name="httpsPort" value="9663"/>
  <property name="httpIdleTime" value="30000"/>
  <property name="httpsIdleTime" value="30000"/>
</bean>
```

4. Edit the following file:

   `/opt/OV/OMU/adminUI/conf/config.properties`

   Change the hostname to `localhost` so the configuration block looks as follows:

   ```
   vendor = Hewlett-Packard Development Company
   backend = rhel-support_server
   hostname = localhost
   server.port = 9661
   ```

5. Restart the Administration UI by running these commands:

   ```
   /opt/OV/OMU/adminUI/adminui clean
   ```

   ```
   /opt/OV/OMU/adminUI/adminui start
   ```

   If an Administration UI patch is applied, this modification must be applied once again.

Caution: Make sure to inform the users about the downtime.

Changing the JMX Port

The JMX port is used only locally for troubleshooting purposes and by some Ant scripts.

The port can be changed as follows:

1. Use one of the following configuration files:
   - `servicemix.properties`

     ```
     # vi /opt/OV/OMU/adminUI/conf/servicemix/servicemix.properties
     [...]
     rmi.port                  = 9660
     rmi.host                  = apollo
     ```

   - `config.properties`

     ```
     # vi /opt/OV/OMU/adminUI/conf/config.properties
     [...]
     # JMX
     jmx.port = 9660
     jmx.user = ${backend.user}
     ```

2. Restart the Administration UI by running the following commands:

   ```
   /opt/OV/OMU/adminUI/adminui clean
   ```

   ```
   /opt/OV/OMU/adminUI/adminui start
   ```

Caution: Make sure to inform the users about the downtime.

Resetting the Default Password for the admin User

After the first logon to the Administration UI, the user is asked to change the default password of the `admin` user. If you forget this password, you can reset it to its initial value (that is, `secret`) by running the following command:

```
/opt/OV/OMU/adminUI/adminui ant -f conf/ant/admin.xml reset_admin_password
```

The reset operation does not affect any other existing user.

The screen output looks as follows:

```
Buildfile: conf/ant/admin.xml
reset_admin_password:
     [echo] Resetting password of admin user
[xdb:extract] Extracting resource: users.xml to /opt/OV/OMU/adminUI/work/users_
20090327141005.xml
[xdb:store] Database driver already registered.
     [echo] Re-login as user admin, password secret
BUILD SUCCESSFUL
Total time: 3 seconds
```

Note: No application restart is necessary.

Switching Between HTTP and HTTPS Communication

Communication on port 9661 between the WebApp and the BackEnd uses HTTP by default. If you want to switch communication to HTTPS or vice versa, the endpoints and certificates must be recreated. Make sure the Administration UI is up and running, and then run the following command:

```
/opt/OV/OMU/adminUI/adminui ant -f conf/ant/admin.xml backend_convert
```

After you recreate the endpoints, exchange and import the HTTPS certificates. For details, see "Using HTTPS" on page 460.

Caution: Known issue in version 4.2.0 (other versions are not affected):

When switching between HTTP and HTTPS, the search index module does not get updated. Therefore, it is necessary to additionally run the following command on the Web Application system:

```
/opt/OV/OMU/adminui ant -f run.xml intern.solr_http
```

Note: To check if secure or unsecure HTTP communication is used, run the following command:

```
/opt/OV/OMU/adminUI/adminui backend
```

No restart of the Administration UI is necessary.

Reinitializing the XML Database

The `adminui init` command can be used to reinitialize the XML database.

Caution: Do not use the `adminui init` command unless advised by Product Support. It is also highly recommended to run the `adminui backup` or `adminui save` command before running the `adminui init` command so that the existing user database is backed up.

The `adminui init` command is needed only under very special circumstances in which the XML database was not correctly initialized (right after an installation).

You can run this command in two different ways:

- `/opt/OV/OMU/adminUI/adminui init`
  Upload and reload of missing parts inside the XML database takes place.
- `/opt/OV/OMU/adminUI/adminui init force`
  This option is used if the XML database setup failed after an installation. The whole XML database is reset (all existing user, user groups, and user roles are lost). Therefore, the initial log-on credentials are as follows:
  - username: `admin`
  - password: `secret`

Note: No restart of Administration UI is necessary.

HPOM Integration

To enable HPOM integration, it is necessary to configure self-monitoring.

This section covers the following two topics:

- "Self-Monitoring" on page 441
- "HPOM Java GUI" on page 442

Self-Monitoring

The Administration UI comes with a set of policies and tools as well as node groups, profiles, and so on, that are all intended for self-monitoring. To use them in an HPOM environment, the following tasks are required:

1. Assign the Administration UI server to the corresponding node groups.

   With the node group assignment, the applicable policies are assigned.
2. Deploy the policies and scripts to the target node (the HP Operations management server). Follow these steps:
   a. Go to Deployment, and then select **Deploy Configuration**.
   b. Select Policies and Monitors as components.
   c. Click the **Please Select** button to open the selector.
   d. From the drop-down menu Locate, select **Node Groups**, and then type **midas** as part of the name.
   e. Double-click the `midas_servers` and `midas_webapps` entries so that they are moved to the selection window.
   f. Close Selector.
   g. Click **Distribute** on the right-hand side.
3. Make the alarm messages available to all HPOM operators. To do so, assign the profile with the name `midas_profile` to your users. Follow these steps:
   a. Select **Assign Profiles**.
   b. Type **midas** as part of the name.
   c. Select `midas_profile` so that it is highlighted.
   d. Finish operation by clicking **OK**.

After you perform these steps, reload the HPOM Java GUI sessions.

Note: If any of the initial self-monitoring policies, scripts, or tools is deleted, there is an easy solution to restore them. All configuration data for the self-monitoring setup and the HPOM Java GUI integration are stored in the following directory:

```
/opt/OV/OMU/adminUI/data/init/ovo/selfmon
```

You can use the `opccfgupld` command to upload this data. For example:

```
cd /opt/OV/OMU/adminUI/
opccfgupld -add $PWD/data/init/ovo/selfmon
```

HPOM Java GUI

In addition to the classic HP Operations Smart Plug-in capabilities, the Administration UI provides a set of HPOM tools that can be also used in the HPOM Java GUI. They all launch the GUI as an HTML page inside the Java GUI and are able to directly jump to the desired context (for example, to the policy condition that caused a message or the node where the message originated).

To use this feature, make sure that the HPOM users have `midas_profile` assigned (see "Self-Monitoring" on page 441).

The Administration UI functions are available through the pop-up menu in the Java GUI. You first select **Start > OMU Administration UI**, and then the desired function.

Caution: The first time you use any of these functions during your session, you are asked to provide your Administration UI username and password.

Configuring Administration UI

This section contains the information about modifying the Administration UI environment.

During the installation, all relevant set-up parameters are defined and stored in various Administration UI configuration files. However, if the HPOM environment changes after the initial installation of the Administration UI, make sure that the changes do not have an adverse effect on the Administration UI. For example, if you change the password for the HPOM administrator, `opc_op`, you must update the Administration UI as well.

The information in this section covers the following topics:

Changing Passwords

The Administration UI comes with several default users (and their equivalent default passwords). In addition, some modules have either a default password set or it is defined during the installation.

This section explains how to change these passwords by using the Administration UI commands.

To reset the password of the Administration UI `admin` user, see "Resetting the Default Password for the admin User" on page 440.

Default Passwords

For a complete list, see "Default Passwords" on page 411.

Password Tool

For security reasons, all passwords stored in the Administration UI are encrypted. To change a password (for example, the database user `opc_op`), use the `adminui` password command:

```
/opt/OV/OMU/adminUI/adminui password <options>
```

The full syntax is as follows:

```
/opt/OV/OMU/adminUI/adminui password -u <useralias> [-a][-c][-p <password>]
```

The parameters are as follows:

`-u <useralias>`	It is mandatory to specify a user alias name to select which password is to be encrypted and updated.
`-a`	Used for updating the password in the corresponding configuration files, so no manual update is needed.
`-c`	Used for checking the password and verifying the entered password against the password used in different Administration UI configuration files. If they match, you receive the message that the password is the same. This feature is useful if you are not sure if the password was updated in the Administration UI.
`-p <password>`	Used for providing a new password.

The `-a` and `-c` parameters cannot be used together because you cannot check and update the password at the same time.

To receive a list of all existing user aliases, run the following command (without using any additional parameters):

```
/opt/OV/OMU/adminUI/adminui password
```

The following users exist:

Alias	Description
`ovodb`	HPOM database user password (`opc_op` or `opc_rep`)

Note: If you encounter a start-up problem in the Administration UI after changing a password inside HPOM but without changing it in the Administration UI, follow these steps:

1. Run the `adminui clean` command:

   `/opt/OV/OMU/adminUI/adminui clean`
2. Change the password.
3. Run the `adminui start` command:

   `/opt/OV/OMU/adminUI/adminui start`

To update the Administration UI configuration with a new `opc_op` password without updating the Administration UI configuration files manually but automatically, run the following command:

`/opt/OV/OMU/adminUI/adminui password -u ovodb -a -p <password>`

Note: If the Administration UI server does not start and the database as well as HPOM are running fine, check the password settings. For more information about typical error codes for problems related to incorrect passwords, see "Problems with Passwords" on page 445.

Problems with Passwords

If the Administration UI does not start or does not work correctly, HPOM access parameters (particularly passwords) might be incorrect. It is required to provide a user and its password that can be used for read-only access to the database during the installation of the Administration UI. If the password is incorrect or the password of the user is changed in the database after the installation without changing it also inside the Administration UI, all HPOM object class listings fail.

To find out how to solve problems related to missing or incorrect passwords, see the following sections:

- "Testing a Password" on page 445
- "Resetting a Password" on page 446
- "Identifying Password Errors" on page 447
- "Oracle Password Aging" on page 448

Testing a Password

If the required password is unknown, in some cases it is possible to connect to the database with an administrator account and reset the password for the HPOM database connection user. If you do not have access to such an account, contact your database administrator to obtain it.

Note: If this is not possible and you want to guess the password by trial and error, keep in mind that it is most probable that the database user account gets blocked after a small number of failed

attempts.

To try to connect to the database, first determine the parameters for the HPOM database connection. For example, run the following command:

```
cat /etc/opt/OV/share/conf/ovdbconf
```

In the `ovdbconf` file, you can find most of the connection parameters you need (that is, the database binary path, the operating system user, the database user, the database name, the host, and the port). Log on to the system with the obtained operating system user name, run a database client tool such as `sqlplus` for Oracle or `psql` for PostgreSQL, and then try to connect to the database with the database user you obtained from the `ovdbconf` file and the password you think is correct. If the connection is successful, the password is correct. Make sure that you also update the Administration UI configuration.

Resetting a Password

Depending on your database, see one of the following two sections:

- "Resetting an Oracle Password" on page 446
- "Resetting a PostgreSQL Password" on page 447

Resetting an Oracle Password

If it is not possible to find out the Oracle password, you must change it.

Caution: You should change the Oracle password only if absolutely necessary. For more information about changing the password, see the opcdbpwd manual page.

To change the Oracle password, follow these steps:

1. Back up the following file:

   `/etc/opt/OV/share/conf/OpC/mgmt_sv/.opcdbpwd.sec`
2. Change the `opc_op` password in HPOM by running the following command:

   `/opt/OV/bin/OpC/opcdbpwd -set`

The `opcdbpwd` command also updates the HPOM internal security file, `opcdbpwd.sec`, with the new authentication that is essential for HPOM to continue to work properly after the password change. If you use the `opcdbpwd` command to change the Oracle password, make sure that you also update the Administration UI configuration.

The `opc_report` password cannot be changed by using `opcdbpwd`. Instead use the required sqlplus commands as shown in the following example:

```
SQL> alter user opc_report identified by <new password>;
SQL> commit;
```

Resetting a PostgreSQL Password

If it is not possible to find out the PostgreSQL password, you must change it.

Caution: You should change the PostgreSQL password only if absolutely necessary.

To change the PostgreSQL password, follow these steps:

1. Back up the `.pgpass` file in the home directory of the operating system DBA user (usually `postgres`).
2. Log on to the PostgreSQL database as the `admin` user (for example, `postgres`).
3. Change the PostgreSQL password by running the following command:

   ```
   postgres=# ALTER USER <user> WITH ENCRYPTED PASSWORD '<password>';
   ```

   Note: If you do not have access to the `admin` user, see the PostgreSQL documentation describing the `pg_hba.conf` file and how to temporarily disable authentication.
4. Edit the `.pgpass` file in the home directory by replacing the old password with the new one, so that HPOM connects to the database with the new password. For details, see the PostgreSQL documentation.

Identifying Password Errors

If the `opc_op` password is incorrect, the Administration UI starts up, but viewing any HPOM object results in an error message being displayed. Password problems related to HPOM and the database connection can be found in the following file:

```
/opt/OV/OMU/adminUI/logs/ovo.log
```

If the password of the database user `opc_op` is incorrect, you receive the following error when trying to access HPOM inside the Administration UI (for example, when requesting to list all nodes or policies):

Inside the `ovo.log` file, the error looks as follows:

```
ERROR - 2008-01-14 21:21:06,768 |
OVODBServer.getConnection(240) | Failed to get connection from
pool ovoconfig java.sql.SQLException: ORA-01017: invalid
username/password; logon denied.
```

Caution: When starting up, the Administration UI does not try to connect to the database. Only when a user actually requests through the WebApp interface a listing of nodes, policies, and so on, the connection is established. Therefore, this error can only be seen after the user actually tried to view some HPOM items through the Administration UI.

Oracle Password Aging

Oracle has password aging enabled by default. This means that passwords expire after 6 months.

If the password of the Oracle user that HPOM uses to connect to the database expires, HPOM cannot connect to the database.

Note: Earlier Oracle versions do not have password aging enabled by default.

To prevent any problems related to password aging, choose one of the following:

- Define a new password every 6 months.

  See "Defining a New Password" on page 448.
- Disable the password aging mechanism.

  See "Disabling the Password Aging Mechanism" on page 448.

Defining a New Password

To define a new password, follow these steps:

1. Change the password in the Oracle database and HPOM by running the following command:

   ```
   /opt/OV/bin/OpC/opcdbpwd -set
   ```
2. Update the Administration UI with the new password:

   ```
   /opt/OV/OMU/adminUI/adminui clean
   /opt/OV/OMU/adminUI/adminui password -u ovodb -a -p <password>
   ```
3. Restart the HP Operations management server processes to make sure the processes that were started before changing the password use the new password:

   ```
   ovc -kill
   ovc -start
   ```

Disabling the Password Aging Mechanism

To disable password aging, run the following commands:

```
su - oracle
$ sqlplus /nolog
SQL> connect / as sysdba
SQL> ALTER PROFILE default LIMIT PASSWORD_LIFE_TIME UNLIMITED;
```

You can verify the status as follows:

```
SQL> select USERNAME, ACCOUNT_STATUS, LOCK_DATE, EXPIRY_DATE, CREATED
SQL> from dba_users;
```

```
USERNAME ACCOUNT_STATUS LOCK_DATE EXPIRY_DATE CREATED
------------------------------ ----------------------
OUTLN OPEN 19-APR-10 21-OCT-09
OPC_REPORT OPEN 19-APR-10 21-OCT-09
OPC_OP OPEN 19-APR-10 21-OCT-09
SYS OPEN 19-APR-10 21-OCT-09
SYSTEM OPEN 19-APR-10 21-OCT-09
DBSNMP EXPIRED & LOCKED 21-OCT-09 21-OCT-09
TSMSYS EXPIRED & LOCKED 21-OCT-09 21-OCT-09
DIP EXPIRED & LOCKED 21-OCT-09 21-OCT-09
ORACLE_OCM EXPIRED & LOCKED 21-OCT-09 21-OCT-09
```

Accessing HPOM and the Database

The information in this section enables you to set up access to both HPOM and the database.

If any of the existing database settings is changed, it is also important to update the corresponding configuration entries inside the Administration UI. Otherwise, the Administration UI cannot connect to the database and any HPOM object class listing request fails (for example, listing the policy bank).

The following database changes affect the Administration UI:

- Hostname change of the remote database server
- Change of the database port
- Change of the database SID
- Database communication method is changed: secure (OCI) or unsecure (thin client)

If any of these configuration parameters is changed, it is also necessary to update the Administration UI.

All database-related information is stored in the following three configuration files:

`/opt/OV/OMU/adminUI/conf/ovoappl.properties`

`/opt/OV/OMU/adminUI/conf/ovoconfig.properties`

`/opt/OV/OMU/adminUI/conf/ovoinstall.properties`

Each of these properties files contains a URL that defines the relevant database connectivity information.

For example:

- For Oracle:

  `ovodb.url=jdbc:oracle:thin:@avocado.hp.com:1521:openview`

  The syntax and fields that could require modification are as follows:

  ovodb.url=jdbc:oracle:*<thin|oci>*@*<Oracle_host>*:*<port>*:*<SID>*

In this instance, *<Oracle_host>* is the Oracle server hostname that hosts the HPOM database, *<port>* is the Oracle port, *<SID>* is the HPOM Oracle database instance name, and *<thin|oci>* is the type of communication Oracle uses (`thin` being unencrypted communication and `oci` being encrypted communication).

- For PostgreSQL:

  `ovodb.url=jdbc:Postgresql://avocado.hp.com:5433/openview`

  The syntax and fields that could require modification are as follows:

  ovodb.url=jdbc:Postgresql://*<PostgreSQL_host>*:*<port>*/*<DB_name>*

  In this instance, *<PostgreSQL_host>* is the PostgreSQL server hostname that hosts the HPOM database, *<port>* is the PostgreSQL port, and *<DB_name>* is the HPOM PostgreSQL database instance name.

Note: For more information about JDBC database connectors as used by the Administration UI, see the following URLs:

- http://www.oracle.com/technetwork/database/application-development/index-099369.html
- http://jdbc.postgresql.org/download.html

Database Connectivity Check

Depending on your database, see one of the following two sections:

- "Oracle Connectivity Check" on page 450
- "PostgreSQL Connectivity Check" on page 452

Oracle Connectivity Check

The type of communication used and the database configuration settings can be checked manually by using one of the following commands:

- `$ORACLE_HOME/bin/tnsping <oracle_server>`

  An output similar to the following one appears:

  ```
  [...](DESCRIPTION=(CONNECT_DATA=(SERVICE_NAME=avocado.hp-intern.com))(ADDRESS=
  (PROTOCOL=TCP)(HOST=192.168.123.123)(PORT=1521)))
  ```

- `$ORACLE_HOME/bin/lsnrctl status`

  An output similar to the following one appears:

  ```
  [...] Listening Endpoints Summary...
    (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(KEY=openview)))  (DESCRIPTION=(ADDRESS=
  (PROTOCOL=tcp)(HOST=avocado.hp-intern.com)
  (PORT=1521)))
  Services Summary...
   [...]
  ```

In these instances, (PROTOCOL=TCP) stands for unencrypted communication and (PROTOCOL=ICP) indicates that you need to enable thin client inside the mentioned configuration files of the Administration UI.

If ICP and TCP are both available, it is recommended to use an unsecure Oracle connection.

If only secure Oracle communication is allowed but standard communication through TCP/IP should be added, this can be achieved by modifying the listener.ora file on the target system. For example:

BEFORE

```
===
LISTENER =
  (ADDRESS_LIST =
        (ADDRESS=
          (PROTOCOL=IPC)
          (KEY= openview)
        )
  )
STARTUP_WAIT_TIME_LISTENER = 0
CONNECT_TIMEOUT_LISTENER = 10
LOG_DIRECTORY_LISTENER = /appl/ora/product/10.1.0/network/log
LOG_FILE_LISTENER = listener
SID_LIST_LISTENER =
  (SID_LIST =
    (SID_DESC =
      (SID_NAME=openview)
      (ORACLE_HOME=/appl/ora/product/10.1.0)
    )
  )
```

AFTER

```
LISTENER =
  (ADDRESS_LIST =
        (ADDRESS=
          (PROTOCOL=IPC)
          (KEY= openview)
        )
                (ADDRESS =
                  (PROTOCOL = TCP)
                  (HOST = abcdefg1)
                  (PORT = 1521)
                )
  )
STARTUP_WAIT_TIME_LISTENER = 0
CONNECT_TIMEOUT_LISTENER = 10
LOG_DIRECTORY_LISTENER = /appl/ora/product/10.1.0/network/log
LOG_FILE_LISTENER = listener
SID_LIST_LISTENER =
  (SID_LIST =
```

```
    (SID_DESC =
      (SID_NAME=openview)
      (ORACLE_HOME=/appl/ora/product/10.1.0)
    )
  )
```

PostgreSQL Connectivity Check

The type of communication used and the database configuration settings can be checked manually by running the following commands:

```
su - postgres
psql -U <DB_DBA_user> -h <hostname> -p <port> -d <database>
```

In this instance, *<DB_DBA_user>* is the name of the administrator user inside the database cluster or server, *<hostname>* is the system on which the database cluster or server is installed, *<port>* is the port on which the database cluster or server listens, and *<database>* is the database name.

You can get the required parameters from the following file:

```
/etc/opt/OV/share/conf/ovdbconf
```

Configuring the Administration UI to Run in a Non-root Operation

To run the Administration UI as a non-root user you must configure the non-root user `opc_op` (enable `opc_op` to start processes) and make sure it belongs to the special `opcgrp` group. Follow one of the following scenarios:

- If the Administration UI is not installed yet, change the file permissions as follows:

  ```
  find /opt/OV/OMU/adminUI/ -type d -exec chmod ug+rw {} \;
  find /opt/OV/OMU/adminUI/ -type f -exec chmod ug+rw {} \;
  ```

- If the Administration is already installed, the client uses the current installation. Follow these steps:

  a. Add the following lines in the `wrapper.conf` file:

  ```
  wrapper.java.umask=0002
  wrapper.logfile.umask=0002
  ```

  b. Perform the following commands:

  ```
  find /opt/OV/OMU/adminUI/ -type d -exec chmod ug+rw {} \;
  find /opt/OV/OMU/adminUI/ -type f -exec chmod ug+rw {} \;
  ```

After the configuration all files and folders should have the `opcgrp` group ownership in addition to the permissions set in these procedures.

Note: All other non-root users must be added to the `opcgrp` group so that they can run the Administration UI commands.

For more information on non-root operation in HPOM, see the *HPOM Concepts Guide*.

Logging and Tracing Mechanism

Both logging and tracing are covered by the same logging mechanism, `log4j`. For details, see the following URL:

http://logging.apache.org

Caution: When changing log levels, no application restart is necessary. The `log4j.xml` file is read every 60 seconds.

General logging for the Administration UI server is configured in the following file:

```
cat /opt/OV/OMU/adminUI/conf/log4j.xml
```

```
[...]
Log4J Configuration Quick Reference:
====================================
Priority order is DEBUG < INFO < WARN < ERROR < FATAL
PatternLayout conversion characters:
%c   Category of the logging event
%C   Fully qualified class name of the caller
[...]
  <!-- MIDAS adaptor default log file -->
  <appender name="midas" class="org.apache.log4j.RollingFileAppender">
    <param name="File" value="logs/midas.log"/>
    <param name="MaxFileSize" value="1MB"/>
    <param name="MaxBackupIndex" value="10"/>
    <layout class="org.apache.log4j.PatternLayout">
      <param name="ConversionPattern" value="%p - %d | %C{1}.%M(%L) | %m%n"/>
    </layout>
  </appender>
[...]
<!-- MIDAS adaptor specific log files -->
<appender name="ovo" class="org.apache.log4j.RollingFileAppender">
[...]
<!-- apache stuff -->
<logger name="org.apache.servicemix" additivity="false">
    <level value="INFO"/>
    <appender-ref ref="servicemix"/>
  </logger>
[...]
<!-- MIDAS adpators -->
  <logger name="com.bes.itm.comp.servicemix" additivity="false">
    <level value="DEBUG"/>
```

```
    <appender-ref ref="midas"/>
  </logger>
[...]
<!-- auditing -->
  <logger name="com.bes.itm.comp.servicemix.DORequestAuditTransformer"
additivity="false">
    <level value="INFO"/>
    <appender-ref ref="audit"/>
  </logger>
[...]
  <!-- default -->
  <root>
    <level value="INFO"/>
    <appender-ref ref="midas"/>
  </root>
</log4j:configuration>
```

The <appender> tag describes the actual log target (for example, the log file name, the entry format, rolling behavior, and so on).

The <logger> tag defines which appender to use with which log level for an Administration UI component.

If you want to configure the log level, change the required <logger> tag. To change the log file behavior, change the <appender> tag.

The following is a list of the trace level settings you can choose in the log files:

Log level	Description
DEBUG	Most detailed level of logged information
INFO	Detailed logging level to report minor and major error conditions, warnings, as well as correct system behavior
WARN	Detailed logging of all unusual warning and error conditions
ERROR	Level to report only error conditions. Warnings and correct system behavior is not logged
FATAL	Only critical system failures are logged

Caution: Setting the log level to NONE disables error logging. It is strongly recommended to set the log level to INFO or at least to WARN.

Web Application Logs

To configure additional logging of the Administration UI Web Application, use the `logkit.xconf` file:

```
cat /opt/OV/OMU/adminUI/webapps/midas/work/webapp/WEB-INF/logkit.xconf
```

```
[...]
  <targets>
  [...]
    <!--
      This log file gets only messages with log level ERROR and below.
    -->
    <priority-filter id="error" log-level="ERROR">
      <cocoon>
        <filename>${context-root}/WEB-INF/logs/error.log</filename>
        <format type="cocoon">          %7.7{priority} %{time}   [%{category}] (%
{uri}) %{thread}/%{class:short}: %{message}\n%{throwable}
        </format>
        <append>false</append>
      </cocoon>
    </priority-filter>
    <cocoon id="debug">
      <filename>${context-root}/WEB-INF/logs/debug.log</filename>
      <format type="cocoon">          %7.7{priority} %{time}   [%{category}] (%
{uri}) %{thread}/%{class:short}: %{message}\n%{throwable}
      </format>
      <append>false</append>
    </cocoon>
[...]
```

Note: The WebApp uses Apache logkit.

Tracing the Administration UI Users

To trace the Administration UI users, follow these steps:

1. Edit `/opt/OV/OMU/adminUI/conf/log4j.xml` file by changing the following part:

   ```
   <logger name="com.bes.itm.comp.xmldb" additivity="false">
     <level value="INFO"/>
     <appender-ref ref="xmldb"/>
   </logger>
   ```

   to this one:

   ```
   <logger name="com.bes.itm.comp.xmldb" additivity="false">
     <level value="TRACE"/>
     <appender-ref ref="xmldb"/>
   </logger>
   ```

2. Restart the Administration UI as follows:

   ```
   /opt/OV/OMU/adminUI/adminui stop
   ```

   ```
   /opt/OV/OMU/adminUI/adminui start
   ```

3. Check the `/opt/OV/OMU/adminUI/logs/xmldb.log` file.

Auditing

Apache `log4j` is used for logging the audit entries. Therefore, all configuration regarding auditing must be done by modifying `log4j` configuration files. For details, see "Logging and Tracing Mechanism" on page 453.

Auditing is enabled by default. The following log levels are available:

- `INFO`

  One line per operation representing a summary of who did what. However, not all details are tracked.

- `DEBUG`

  Full internal requests and responses exchanged of an operation are logged.

The following is an example of an audit record with the `INFO` level:

```
INFO,2009-04-07 12:44:48,168,modifyresponse,1239101088110, 5fnw9g49a0,tge,Web
UI,avocado_server,ovoconfig,modify,ovo:policy, ,ok,,,,,,1,,,Bad Logs (11.x HP-
UX),logfile,2.0,,,,false,false,true
```

In this instance, the `tge` user modified the log file policy named `Bad Logs (11.x HP-UX)`on the `avocado_server` BackEnd.

With the `DEBUG` level set inside `log4j.xml`, the full data flow is captured and logged (that is, all the details about the objects being modified). To enable the `DEBUG` mode, change the `log4j.xml` file as follows:

```
[...]
  <!-- auditing -->
  <logger name="com.bes.itm.comp.servicemix.DORequestAuditTransformer" additivit
y="false">
    <level value="DEBUG"/>
    <appender-ref ref="audit"/>
  </logger>
[...]
```

The following is an example of an audit record with the `DEBUG` level:

```
DEBUG,2009-04-07 12:59:57,171,<modifyresponse ...
  [...]
  <backend do:type="String" xml:space="preserve">avocado_server</backend>
  <operation do:type="String" xml:space="preserve">modify</operation>
  <objectclass do:type="String" xml:space="preserve">ovo:policy</objectclass>
  <user do:type="String" xml:space="preserve">tge</user>
```

```
  <version do:type="String" xml:space="preserve">2.1</version>
[...]
  <objectname do:type="String" xml:space="preserve">Bad Logs (11.x HP-UX)
</objectname>
  [...]
```

The audit files are written in `/opt/OV/OMU/adminUI/logs/audit`. The active log file is `audit.log`. By default, this file is archived daily to files such as `audit.log.2009-03-20`. This behavior can be controlled by editing `/opt/OV/OMU/adminUI/conf/log4j.xml`:

```
<appender name="audit" class="org.apache.log4j.DailyRollingFileAppender">
 <param name="File" value="logs/audit/audit.log"/>
 <param name="DatePattern" value="'.'yyyy-MM-dd"/>
 <layout class="org.apache.log4j.PatternLayout">
 <param name="ConversionPattern" value="%p,%d,%m%n"/>
 </layout>
</appender>
[...]
<logger name="com.bes.itm.comp.servicemix.DORequestAuditTransformer"
 additivity="false">
 <level value="DEBUG"/>
 <appender-ref ref="audit"/>
</logger>
```

The Administration UI does not provide any tools to review the audit records. However, the audit log files can be loaded as CSV files into spreadsheet applications such as MS Excel or OpenOffice for evaluation.

If needed, select a comma as a separator.

Furthermore, there is also a formatted audit output available that can be viewed inside the Administration UI web interface. This function can be accessed through the context of the Server icon where you select **Browse > Formatted Audit**.

The following is a list of all audit fields:

Audit Field	Description
`log level`	Log level (usually INFO)
`log date`	Time stamp when an entry was logged.
`request type`	Type of a request or response (`getrequest`, `getresponse`, `listrequest`, `listresponse`, and so on).
`timestamp`	Time stamp in Unix time when the request was created.
`uid`	Unique ID of the request that can also be found in the response or follow-on requests.

Audit Field	Description
user	User name that sent the request.
sender	Module form which the request came.
backend name	Backend identifier the request is being routed to.
service name	Target adaptor name the request is being routed to.
operation	Performed operation (get, list, create, modify, delete, assign, and so on).
object class	Object class handled with the request. Mass requests have an object class name all with a namespace prefix.
context	Context
status	Status of the response operation.
details mode	Details mode that enables the modification of the output format of the get and list operations, and do on.
force flag	Flag often used to enforce an operation (for example, caching, backend resolution, lock override, and so on).
comment	Optional text entered in the comment field.
version	Version in a version control system

Depending on the type of request, further attributes may be logged such as objectname, objecttype, contextobjectname, or flags such as recursive all or inherited.

Request Logging Mechanism

Overview

The request logging mechanism is used for logging every operation in detail. In addition, all the internal requests that are considered as unloggable, can be logged by using this mechanism.

The request logging mechanism is disabled by default because it requires a considerable amount of disk space. Therefore, when enabling it, check your disk space for /opt/OV/OMU/adminUI/logs/requests.

Note: This type of logging mechanism is recommended only for debugging.

The request logging mechanism can be enabled for the following:

- audit request logging: typical user actions (for example, edit, add, remove requests)
- internal request logging: internal server requests

Setup

Besides the standard `audit.log`, it is also possible to enable the detailed request logging mechanism to log the user actions. Enabling this mechanism is recommended only for troubleshooting purposes because it affects the used disk space and CPU consumption.

You can enable the request logging mechanism in the following file:

`/opt/OV/OMU/adminUI/conf/becore.properties`

In this case, `false` disables request logging, whereas `true` enables it.

The contents of `becore.properties` is:

```
layer config file for backend documentor features
request auditing (incoming requests/responses)
auditing = false
request logging (authenticated or internal requests/responses)
logging = false
eof
```

The following parameters can be modified:

- `auditing`

  Typical user actions (for example, edit, add, remove requests)
- `logging`

  This refers to the internal request logging for those items otherwise not logged during internal Administration UI operations.

`audit.log` lists all external requests, whereas `request.log` (when enabled) lists adapter-specific internal requests.

Advanced Communication Options

This section describes how to set up advanced communication options to improve the performance of the installed software. The information covers the following topics:

- "Changing Default Ports" on page 460
- "Using HTTPS" on page 460
- "Using Proxies" on page 473
- "Using the Administration UI in Firewall Environments" on page 473

Changing Default Ports

The communication relationships and default ports used by the Administration UI are explained in "Communication and Ports" on page 404.

Most ports are defined during the installation.

Caution: If you modify port settings, make sure to perform a clean restart of the Administration UI.

You must also make sure to inform the users about the downtime.

If it is necessary to change any port, hostname, or identifier, see one of the following sections:

- "Renaming the BackEnd Identifier" on page 436
- "Changing the Hostname" on page 436
- "Changing the BackEnd Port (9661)" on page 436
- "Changing the WebApp HTTP or HTTPS Port " on page 437
- "Disabling the WebApp HTTP Port (9662)" on page 438
- "Changing the JMX Port" on page 439

Using HTTPS

This section explains how to configure HTTPS to improve the security of communication between the installed components. The information covers the following areas:

- "Understanding HTTPS" on page 460
- "Configuring HTTPS in the Administration UI" on page 461
- "HTTPS Between the Browser and the Web Application" on page 462
- "jetty.xml Configuration File Properties" on page 463
- "Replacing Self-signed Certificates with Custom Certificates" on page 468
- "Configuring Client-Side Certificates" on page 472

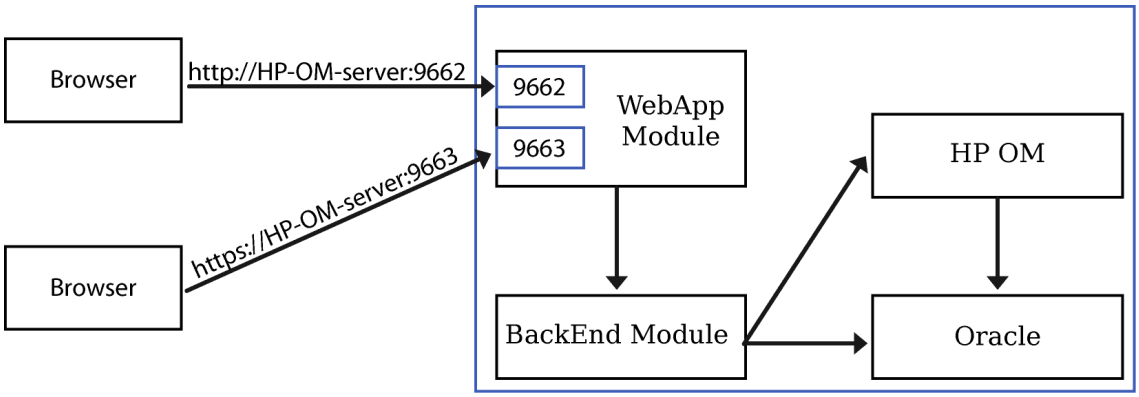
Understanding HTTPS

HTTPS communication provides the following two security-related features in addition to basic HTTP communication:

- Authentication
- Encryption

Figure 18 shows the relationship between HTTP and HTTPS within the Administration UI.

Figure 18: Communication Protocols



The WebApp acts as an HTTP or HTTPS server and the GUI as a client.

These are the roles of HTTPS communication within the Administration UI:

- Authentication
  - Server authentication

    The HTTPS server must provide some credentials that can be used by the client to verify the server's identity. This is mandatory when using HTTPS.

  - Client authentication

    In addition to server authentication, the client must also authenticate itself to the server so that the server can verify the client's identity.

- Authorization

  This is actually not part of HTTPS but can be used to restrict the set of clients that are allowed to perform operations on the server.

It is also conceivable and useful for the Administration UI communication to restrict access to only a specific set of users.

Configuring HTTPS in the Administration UI

The following features of HTTPS communication can be used within the Administration UI:

- server authentication only
- client authentication (optional, disabled by default)

By default, the Administration UI generates self-signed certificates on all Administration UI servers that can be used for out-of-the-box HTTPS communication. If this is an acceptable security level, nothing else needs to be done. Otherwise, certificates may need to be generated and imported to the keystores and truststores.

Note: You can replace self-signed certificates with custom certificates. For more information on how to do this, see "Replacing Self-signed Certificates with Custom Certificates" on page 468.

The keystores and truststores used by the Administration UI server can be found in the following directory:

```
ls -l /opt/OV/OMU/adminUI/conf/servicemix/*store*
```

```
-rw-rw-r-- 1 root root 2245 May 12 13:21
/opt/OV/OMU/adminUI/conf/servicemix/keystore_webapp.jks
-rwxrwxr-x 1 root root 3243 Apr 28 18:03
/opt/OV/OMU/adminUI/conf/servicemix/keystore.jks
-rw-rw-r-- 1 root root  956 May 12 13:21
/opt/OV/OMU/adminUI/conf/servicemix/truststore_webapp.jks
-rwxrwxr-x 1 root root  662 Apr 28 18:03
/opt/OV/OMU/adminUI/conf/servicemix/truststore.jks
```

The files named `*_webapp.jks` are related to the Administration UI HTTPS communication. All following references to keystores and truststores are related to these files.

The other files are required by infrastructure components (Jetty Web container and ServiceMix). Make sure not to modify these files.

Authorization within the Administration UI is controlled entirely by the Administration UI user model. Therefore, there is nothing to be configured elsewhere. If an HTTPS client is able to establish an HTTPS connection (and the certificate exchange is successful), it is also considered as authorized to perform all tasks.

HTTPS Between the Browser and the Web Application

For out-of-the-box server-side authentication, the user must only accept the WebApp certificate in the browser. Web browser cannot validate the default self-signed certificate if it does not have the signing CA certificate.

To inspect the certificate, add an exception by clicking **Or you can add an exception...**, and then selecting **Add Exception**. Now you can view the certificate and confirm the security exception.

A warning might appear if there is a mismatch between the server hostname and the common name (CN), which is the name to which the HTTPS server certificates are usually issued. Click OK to proceed.

You can choose whether to accept the certificate permanently or repeat this process again next time. If you accept the certificate permanently, it is placed in the web browser's truststore.

Caution: If the names displayed in the Domain-Name-Mismatch window are not the HPOM hostname and the Administration UI server ID defined during the installation of the Administration UI, it is possible that a real security threat exists. To avoid this problem, the WebApp certificate

must be signed by a certification authority (CA) registered in the web browser.

jetty.xml Configuration File Properties

`jetty.xml` is the main configuration file for Jetty and is located at `/opt/OV/OMU/adminUI/conf/`. It defines several components that are essential for configuring and running the web server. This file enables you to configure the HTTP and SSL connectors by using the following properties:

- HTTP configuration

Property name	Description
`<property name="httpPort" value="9662"/>`	Sets the port on which the application listens for HTTP requests.
`<property name="httpsPort" value="9663"/>`	Sets the port for Confidential and Integral redirections.
`<property name="httpIdleTime" value="30000"/>`	Sets the maximum idle time for an HTTP connection.
`<property name="httpsIdleTime" value="30000"/>`	Sets the maximum idle time for a secure HTTP connection.
`<property name="outputBufferSize" value="32768"/>`	Sets the size of the buffer into which HTTP output is aggregated.
`<property name="requestHeaderSize" value="8192"/>`	Sets the maximum size of a request header.
`<property name="responseHeaderSize" value="8192"/>`	Sets the maximum size of a response header.
`<property name="headerCacheSize" value="512"/>`	Sets the maximum header field cache size.
`<property name="securePort" value="9663"/>`	Sets the port used for Confidential and Integral redirections. It is overridden with the `httpsPort` property.
`<property name="secureScheme" value="https"/>`	Sets the scheme used for Confidential and Integral redirections.
`<property name="sendServerVersion" value="true"/>`	If set to `true`, sends the Server header in responses.
`<property name="sendXPoweredBy" value="false"/>`	If set to `true`, sends the X-Powered-By header in responses.
`<property name="sendDateHeader" value="true"/>`	If set to `true`, includes the date in HTTP headers.

- SSL configuration

  The server key, the keystore, and the truststore are protected with encrypted passwords and have the same default value (that is, `password`).

  The following passwords are stored as properties in the `jetty.xml` configuration file:

Property name	Description
`<property name="keyManagerPassword" value="**********" />`	Used to access the `keystore_webapp.jks` file. If this file is changed, the property for the password must be changed as well.
`<property name="keyStorePassword" value="**********" />`	Used to access the certificate located in the `keystore_webapp.jks` file. If the server key in this file is changed, the property for the password must be changed as well.
`<property name="trustStorePassword" value="**********" />`	Used to access the `truststore_webapp.jks` file. If this file is changed, the property for the password must be changed as well. [a]

[a] Because passwords are encrypted, you must use a password tool to change or check them. For more information about this tool, see "Password Tool" on page 444.

Property name	Description
`<property name="includeProtocols">`	Used to define a list of SSL protocols that a server uses for HTTPS-based communication [a]

Enabling or Disabling SSL Protocols

The `jetty.xml` file defines a list of enabled SSL protocols as follows:

```
...
<list>
   <value>TLSv1</value>
   <value>TLSv1.1</value>
   <value>TLSv1.2</value>
   <value>SSLv3</value>
</list>
...
```

You can enable or disable SSL protocols by commenting out or removing the lines related to them from this list. For example, to disable the `SSLv3` protocol, remove `<value>SSLv3</value>` from the `<list>` node.

For your changes to take effect, you must restart the Administration UI as follows:

[a]You can specify the cipher suites or protocols to be used by the Jetty web server:
<propertyname="includeProtocols">
<list>
<value>TLSv1</value>
<value>TLSv1.1</value>
<value>TLSv1.2</value>
<value>SSLv3</value>
....
</list>
</property>
<propertyname="excludeCipherSuites">
<set>
<value>SSL_RSA_WITH_DES_CBC_SHA</value>
<value>SSL_DHE_RSA_WITH_DES_CBC_SHA</value>
<value>SSL_DHE_DSS_WITH_DES_CBC_SHA</value>
....
</set>
</property>
<propertyname="includeCipherSuites">
<set>
<value>TLS_DHE_DSS_WITH_AES_128_CBC_SHA</value>
<value>TLS_DHE_RSA_WITH_AES_128_CBC_SHA</value>
....
</set>
</property>

```
/opt/OV/OMU/adminUI/adminui clean
/opt/OV/OMU/adminUI/adminui start
```

Changing Passwords by Using the Administration UI Password Tool

To change passwords by using the Administration UI password tool, run the Administration UI password tool located in the `/opt/OV/OMU/adminUI` directory. Choose one of the following methods:

- To change the password automatically (recommended), run the following command:

  ```
  adminui password -u <useralias> -p <password> -a
  ```

- To change the password manually, obtain the encrypted form of the plain-text password by running the following command:

  ```
  adminui password -u <useralias> -p <password>
  ```

  Alternatively, you can use a GUI:

  ```
  adminui password -u <useralias> –i
  ```

Keep in mind that by doing so you obtain only the encrypted version of the password you entered. Therefore, you must manually replace the property value in the `jetty.xml` file with the corresponding encrypted value.

User alias	Property in jetty.xml
keystore	keyStorePassword
trustPassword	trustStorePassword
keyPassword	keyManagerPassword

Checking Passwords by Using the Administration UI Password Tool

You cannot retrieve a plain password from `jetty.xml`, but you can check if your password matches the one in the `jetty.xml` file by using the Administration UI password tool. To do this, run the Administration UI password tool located in the `/opt/OV/OMU/adminUI` directory. Choose one of the following methods:

- To check the password automatically (recommended), run the following command:

  ```
  adminui password -u <useralias> -p <password> -c
  ```

- To check the password manually, obtain the encrypted form of the plain-text password by running the following command:

  ```
  adminui password -u <useralias> -p <password>
  ```

  Alternatively, you can use a GUI:

  ```
  adminui password -u <useralias> –i
  ```

After that, you can compare the obtained password with the corresponding one in the `jetty.xml` file.

Replacing Self-signed Certificates with Custom Certificates

Certificates within the Administration UI can be generated using any desired mechanism (for example, `keytool`, `openssl`, the SSH key generator command, or the HPOM SecCore functionality). Keep in mind that `keytool` is supported by HP. Before you start, make sure that you have the latest Accessories patch installed.

Replacing self-signed certificates that are created during the installation with the custom CA certificates includes the following tasks:

- Task 1: "Backing up the Current Keystore and Truststore" on page 468
- Task 2: "Replacing the Certificates" on page 468

Backing up the Current Keystore and Truststore

Make sure that you back up the current keystore and truststore in a directory outside the Administration UI (for example, the `/tmp` directory). To do so, run the following commands:

```
cp /opt/OV/OMU/adminUI/conf/servicemix/keystore_webapp.jks <directory>
cp /opt/OV/OMU/adminUI/conf/servicemix/truststore_webapp.jks <directory>
```

Replacing the Certificates

To replace self-signed certificates created during the installation with custom CA certificates, follow these steps:

1. Make sure that you have a certificate authority (CA) to sign your certificate.

   Before you can use your newly obtained certificate, you may have to import the CA root certificate into the keystore.

   > **Note:** You can also create your own CA certificate by using the `openssl` command.

   To list all preinstalled root certificates, run the following command:

   ```
   /opt/OV/nonOV/jre/b/bin/keytool -keystore
   /opt/OV/nonOV/jre/b/lib/security/cacerts -storepass changeit -list
   ```

   You can also add the `-v` option at the end to obtain the same list with more details, such as expiration dates.

   If your certificate authority is not listed, or you were notified that new root certificates are used, you must import the CA root certificate into your keystore before importing your newly obtained certificate.

   The CA root certificate is available for the download from the CA web site as a file with a `.pem` or `.crt` extension. Save the file using a file name that indicates the CA name (for example, `CA.crt`).

Choose one of the following methods to import the root certificate:

- Add the CA root certificate to the `/opt/OV/nonOV/jre/b/lib/security/cacerts` file as follows:

  ```
  /opt/OV/nonOV/jre/b/bin/keytool -import -trustcacerts -keystore
  /opt/OV/nonOV/jre/b/lib/security/cacerts -storepass changeit -alias <ca_
  alias> -file <ca_root>.crt
  ```

  When asked whether to trust this certificate, type `yes`.

  You must be very careful what you add something to the `cacerts` file because all certificates that are signed by this CA are trusted.

- Before replacing the self-signed certificate, add the CA root certificate to the keystore and truststore.

  Add the CA root certificate to the `/opt/OV/OMU/adminUI/conf/servicemix/keystore_webapp.jks` file as follows:

  ```
  /opt/OV/nonOV/jre/b/bin/keytool –keystore
  /opt/OV/OMU/adminUI/conf/servicemix/keystore_webapp.jks -storepass <keystore_
  pass> -importcert -alias <ca_alias> -file <ca_root>.crt
  ```

  In this instance, *<keystore_pass>* is the password for `keystore_webapp.jks` (the default is `password`).

  Add the CA root certificate to the `/opt/OV/OMU/adminUI/conf/servicemix/truststore_webapp.jks` file as follows:

  ```
  /opt/OV/nonOV/jre/b/bin/keytool -import -alias ca_crt -keystore
  /opt/OV/OMU/adminUI/conf/servicemix/truststore_webapp.jks -storepass password
  -file <cert_file_name>
  ```

  When asked whether to trust this certificate, type `yes`.

2. Remove all old keys and certificates from the keystore and truststore of the WebApp as follows:
   a. Check the contents of the keystore and the truststore by running the following command:

   ```
   /opt/OV/OMU/adminUI/adminui ant -f run.xml https_list_webapp
   ```

   Note: Note the alias name, as you will need the alias name from the output in the following step.

   b. Remove the certificates by running the following commands:

   ```
   /opt/OV/nonOV/jre/b/bin/keytool -delete -alias <alias_name> -keystore
   /opt/OV/OMU/adminUI/conf/servicemix/keystore_webapp.jks
   ```

   ```
   /opt/OV/nonOV/jre/b/bin/keytool -delete -alias <alias_name> -keystore
   /opt/OV/OMU/adminUI/conf/servicemix/truststore_webapp.jks
   ```

When running these commands, you must provide the correct passwords. The default is `password`. However, if the default password is changed, make sure to use the changed one. For detailed information about how to check if you are using the correct password, see "Checking Passwords by Using the Administration UI Password Tool" on page 467.

c. Check if the certificates are removed from the keystore and truststore, by running the following command:

```
/opt/OV/OMU/adminUI/adminui ant -f run.xml https_list_webapp
```

3. Generate a new server key and certificate by running the following command:

```
/opt/OV/nonOV/jre/b/bin/keytool -keystore
/opt/OV/OMU/adminUI/conf/servicemix/keystore_webapp.jks -storepass <keystore_
pass> -genkey -alias <alias_name> -keyalg RSA -sigalg <sig_alg> -keysize <key_
size> -validity <val_days> -v
```

In this instance, consider the following:

<keystore_ pass>	Default password for `keystore_webapp.jks` is `password`. If the default password is changed, make sure to use the changed one. For detailed information about how to check if you are using the correct password, see "Checking Passwords by Using the Administration UI Password Tool" on page 467.
<sig_alg>	Specifies the algorithm that should be used to sign the self-signed certificate. This algorithm must be compatible with `keyalg` (the default is `SHA1withRSA`).
<key_size>	Default is 1024 bits (the standard is 2048 bits). Make sure that it is not greater than 65536.
<val_days>	Default is `90`.
`-v`	When this option is used, more information is provided in the output.

This command prompts for the information about the certificate and for passwords to protect both the keystore and the keys within it. The only mandatory response is to provide the fully-qualified hostname of the server at the “first and last name” prompt.

For example:

```
Enter keystore password:  password
What is your first and last name?
  [Unknown]: www.hp.com
What is the name of your organizational unit?
  [Unknown]: R&D
What is the name of your organization?
  [Unknown]: Hewlett-Packard Development Company
```

```
What is the name of your City or Locality?
  [Unknown]: Stuttgart
What is the name of your State or Province?
  [Unknown]: Baden-Wuerttemberg
What is the two-letter country code for this unit?
  [Unknown]: DE
Is CN=www.hp.com, OU=R&D, O=Hewlett-Packard Development Company,
L=Stuttgart, ST=Baden-Wuerttemberg, C=DE?
  [no]:  yes

Enter key password for <jetty>
        (RETURN if same as keystore password):  password
```

If you do not want to use a new password, the password will be the same as the one for the keystore. The default password for the keystore is `password` (the same as the default one for the key). In this case, you do not need to change any password in the `jetty.xml` file.

If you enter a new password for the key, you must replace the old password with the new one in the `jetty.xml` file (the value of `keyManagerPassword` must be changed). For detailed information about how to replace the password for the key, see "Changing Passwords by Using the Administration UI Password Tool" on page 467.

4. Generate a certificate signing request (CSR).

   Note: To obtain a certificate that will be trusted by most common browsers, you need to request a certificate authority (CA) to sign your certificate.

   To generate the `signrequest.csr` file by using a keytool for the key/certificate pair that is already in the keystore, run the following command:

   ```
   /opt/OV/nonOV/jre/b/bin/keytool -certreq -alias <alias_name> -keystore
   /opt/OV/OMU/adminUI/conf/servicemix/keystore_webapp.jks -storepass <keystore_
   pass> -sigalg <sig_alg> -file /tmp/signrequest.csr
   ```

<alias_name>	Alias that is used when the key is generated.
<sig_alg>	Algorithm that should be used to sign the CSR (the default is `SHA1withRSA`).
<keystore_ pass>	Password for `keystore_webapp.jks` (the default is `password`). If the default password is changed, make sure to use the changed one. For detailed information about how to check if you are using the correct password, see "Checking Passwords by Using the Administration UI Password Tool" on page 467.

5. Export the private server key by running the following command:

   ```
   /opt/OV/nonOV/jre/b/bin/keytool -importkeystore -keystore
   /opt/OV/OMU/adminUI/conf/servicemix/keystore_webapp.jks -storepass <password> -
   ```

```
destkeystore <PKCS12_FILE>.p12 -deststoretype PKCS12 -srcalias <alias name> -
deststorepass <password> -destkeypass <PASSWORD_FOR_PKCS12_FILE>
```

The CA signs the certificate and sends you the signed certificate.

6. Import the signed certificate into the keystore and truststore.

   If you used the first option for importing the root certificate in step 1, you must add the `-trustcacerts` option to the following commands (because your trust certificate is in the `cacerts` file):

   ```
   /opt/OV/nonOV/jre/b/bin/keytool -import - keystore
   /opt/OV/OMU/adminUI/conf/servicemix/keystore_webapp.jks
   -storepass <keystore_pass> -alias <alias_name> -file <ca_returned>.crt
   ```

   ```
   /opt/OV/nonOV/jre/b/bin/keytool -import -alias <alias name> -keystore
   /opt/OV/OMU/adminUI/conf/servicemix/truststore_webapp.jks
    -storepass <keystore_pass> -file <ca_returned>.crt
   ```

   In this instance, *<keystore_pass>* is the password for `keystore_webapp.jks` and `truststore_webapp.jks` (the default is `password`). When prompted for the password, enter the one that you have used when you generated the key.

   Caution: Make sure you enter the *<alias_name>* that you have used when you generated the key.

7. Load the CA certificate into the browser.

   Add the CA root certificate to your browser as “Trusted Root Certification Authorities”, so the browser can trust the signed certificate in your server’s keystore, and then restart your browser.

8. Restart the Administration UI by running the following commands:

   ```
   /opt/OV/OMU/adminUI/adminui clean
   ```

   ```
   /opt/OV/OMU/adminUI/adminui start
   ```

Configuring Client-Side Certificates

If you have a client certificate, you can configure client-side certificates for the GUI and WebApp relationship:

- If client authentication is required, set the `needClientAuth` property to `true` in the `jetty.xml` file:

  ```
  <bean class="org.mortbay.jetty.security.SslSocketConnector">
      <property name="port" value="9663"/>
      ****
    <property name="needClientAuth" value="true"/>
  </bean>
  ```

  If the `needClientAuth` property does not exist, add it, and then set its value to `true`.

In this case, the server requests the client authentication. If the client does not have the certificate, it cannot access the Administration UI.

- If client authentication is not required, but you still want it to be done, set the `wantClientAuth` property to `true` in the `jetty.xml` file:

```
<bean class="org.mortbay.jetty.security.SslSocketConnector">
    <property name="port" value="9663"/>
    ****
  <property name=" wantClientAuth" value="true"/>
</bean>
```

If the `wantClientAuth` property does not exist, add it, and then set its value to `true`.

In this case, the server requests the client authentication. However, even if the client does not have a certificate, it can still access the Administration UI.

If the client authentication is required, you must import the root certificate of the CA that signed the client certificate into `/opt/OV/OMU/adminUI/conf/servicemix/truststore_webapp.jks` by running the following command:

```
/opt/OV/nonOV/jre/b/bin/keytool -import –keystore
/opt/OV/OMU/adminUI/conf/servicemix/truststore_webapp.jks -storepass <truststore_
pass> -alias <ca_client_alias> -file <ca_client>.pem
```

In this instance, *<truststore_pass>* is the password for `truststore_webapp.jks` (the default is `password`). If the default password is changed, make sure to use the changed one. For detailed information about how to check if you are using the correct password, see "Checking Passwords by Using the Administration UI Password Tool" on page 467.

When asked whether to trust this certificate, answer `yes`.

After that, restart the Administration UI by running the following commands:

```
/opt/OV/OMU/adminUI/adminui clean
```

```
/opt/OV/OMU/adminUI/adminui start
```

Note: When connecting to the Administration UI, you may need to select the correct client certificate (if there is more than one client certificate).

Using Proxies

Using an HTTP proxy between the web browser and the Administration UI WebApp is currently not supported.

Using the Administration UI in Firewall Environments

For details, see "Communication and Ports" on page 404.

Multiple Concurrent Sessions

The Administration UI allows multiple concurrent sessions by default. This means that a user can be logged on to several Administration UI sessions at the same time, either on the same machine (using different browsers) or on different machines.

Preventing a User From Having Multiple Concurrent Sessions

To prevent a user from having multiple concurrent sessions, follow these steps:

1. Navigate to the `auth.properties` configuration file:

   `/opt/OV/OMU/adminUI/conf/auth.properties`
2. Change the value of the `userauth-filter.concurrentSessionsEnabled` property from `true`, which is the default value, to `false`.

When the value of the `userauth-filter.concurrentSessionsEnabled` property is set to `false`, you can customize the following inside the `auth.properties` configuration file:

- Number of concurrent sessions allowed per user

  To limit the number of concurrent sessions allowed per user, use the `userauth-filter.concurrentSessions` property. For example:

  `userauth-filter.concurrentSessions=1`

  This means that only one concurrent session is allowed per user.

  > **Note:** If you set the value of the `userauth-filter.concurrentSessions` property to `0` (default value), the concurrent session functionality is disabled and the value for `concurrentSessionsEnabled` is reset to default.
  >
  > The number of concurrent sessions will be used only if the value of the `concurrentSessionsEnabled` property is set to `true`.

- Inactivity period

  To set the number of minutes after which a user that is inactive for a specified period of time is logged out of the Administration UI, use the `userauth-filter.inactivityTimeout` property. For example:

  `userauth-filter.inactivityTimeout=60`

  This means that a user that is inactive for 60 minutes is automatically logged out of the Administration UI and a logout record is created in the log files.

  > **Note:** If you use the default value (that is, `0`), the inactivity period functionality is disabled. The inactivity functionality is used only if the value of `concurrentSessionsEnabled` property is set

to `false`.

If a user is inactive, the user will be logged out and a logout record is created in the log files.

- Automatic Logout

  To set an automatic logout for a user that reached the specified number of concurrent sessions when a new user with same credentials logs on, set the `userauth-filter.automaticLogout` property to `true` (the default value is `false`).

- Manual Logout

  To log out a user manually, type the following:

  `/opt/OV/OMU/adminUI/adminui logout_user <username> <session_ID>`

Caution: After any modification, the Administration UI must be restarted by running the following command:

`/opt/OV/OMU/adminUI/adminui restart`

Monitoring User Sessions

To monitor the users that are logged on to the Administration UI, use the `listconn` tool located in the following directory:

`/opt/OV/OMU/adminUI/bin/`

The following information about the user is available:

- User name
- Session ID
- Logon time

To log a user out from the Administration UI, run the following command:

`/opt/OV/OMU/adminUI/adminui logout_user <user_name> <session_id>`

Note: The user is not logged out in the same way as when the standard logout action is performed, but on the first action that requires user authorization. A logout record is created in the log files.

Quality of Service Filter

You can use the Quality of Service Filter (QoSFilter) to limit the number of active requests being handled for specific URLs within the Administration UI. Any requests exceeding the defined limit are suspended. When a request completes handling the limited URL, one of the waiting requests resumes and can be handled.

To set the QoSFilter, use the `web.xml` configuration file located in the following directory:

```
/opt/OV/OMU/adminUI/webapps/midas/work/webapp/WEB-INF/
```

Search for the following section:

```
[...]
<filter>
  <filter-name>QoSFilter</filter-name>
  <filter-class>org.mortbay.servlet.QoSFilter</filter-class>
  <init-param>
     <param-name>maxRequests</param-name>
     <param-value>100</param-value>
  </init-param>
</filter>
[...]
```

After any modification, the Administration UI must be restarted by running the following command:

```
/opt/OV/OMU/adminUI/adminui restart
```

Caution: Make sure to inform the users about the downtime.

Account Locking

When a user fails to log on to the Administration UI in the several attempts, the account of this user is locked until it is enabled again by the administrator.

To enable this account again, choose the corresponding action from the action menu of this user in the All Users page of the Administration UI.

To change the default configuration, modify the *<ADMINUI_HOME>*`/conf/usermgmt.properties` file. Consider the following file properties:

- `usermgmt.accountLocking`

  Determines whether account locking functionality is enabled. The default value is `true`.
- `usermgmt.maxLoginRetries`

  Determines how many failed login attempts are allowed per user, before locking. The default value is 6, and the maximum value is 10.

Note: Administrator's account (`admin`) cannot be locked for the security reasons.

In case that logging for user management is enabled in the `log4j.xml` file, warning messages related to the locking of a user account are logged in the application log files (`/opt/OV/OMU/adminUI/logs/usermgmt.log`).

Protecting Against Cross-site Request Forgery (CSRF) Attacks

The Administration UI includes CSRF token-based protection. This prevents a user who is logged on to the Administration UI from causing unwanted modification of data, for example, by opening a malicious web page. Table 37 shows which parameters you can use to customize the CSRF protection. You can find these parameters in the `security.properties` file that is located at:

`/opt/OV/OMU/adminUI/conf/`

Table 37: CSRF-related Parameters

Parameter	Description
`synchronization.token.name`	Name of the synchronization token (that is, the name of the synchronization parameter in the Administration UI requests). The default value is `SYNC_TOKEN`.
`token.per.request`	Specifies if the synchronization token is generated per session (when set to `false`) or per request (when set to `true`). **Caution:** Setting the value of this parameter to `true` may cause navigation problems (for example, the back button may not work properly because this action is understood as a CHRF attack by the request handler). The default and recommended value is `false`.
`number.generator`	Algorithm that is used for the generation of the synchronization token value. For more information about available algorithms, see http://docs.oracle.com/javase/7/docs/api/java/security/SecureRandom.html. The default value is `SHA1PRNG`.
`number.generator.provider`	Provider of the number generator. For more information, see http://docs.oracle.com/javase/7/docs/api/java/security/SecureRandom.html. The default value is `SUN`.
`token.session.key`	Parameter name used when the synchronization token is stored in the user session. The default value is `TOKEN_SESSION_KEY`.
`protect.ajax.requests`	Determines if CSRF checks should be enabled for Ajax requests.

CSRF-related Parameters , continued

Parameter	Description
	The default value is `true`.
`ignore.*`	Resources that should be ignored by the CSRF filter. The resources that should be ignored are matched using standard java URI mapping: • exact match (for example, `ignore.TokenServlet=/midas/TokenServle`) • longest path prefix match, beginning with `/` and ending with `/*` • extension match, beginning with `*` (for example, `ignore.png=*.png`) **Caution:** Deleting some of the preconfigured ignore properties causes the Administration UI CSRF protection to stop working correctly. Therefore, the ignore list can only be extended with new URIs.

Web Interface Timeout

This section lists the configuration steps that are necessary to increase the timeout for the Administration UI web interface. Otherwise, for example, if a user leaves the policy editor window open and returns after three hours, the session timed out.

To increase the timeout, the following three timeout settings must be modified:

- (1) Continuation Timeout
- (2) Backend Session Timeout
- (3) Webapp Session Timeout

These three timeouts are dependent on each other and the timeout must increase from (1) to (3). In other words, the Continuation Timeout (1) is smaller than the Backend Session Timeout (2) that is in return smaller than the Webapp Session Timeout (3). Therefore, the timeout defined for (2) must be greater than the timeout defined for (1), and the timeout defined for (3) must be greater than the timeout defined for (2). Any difference, for example, a minute or an hour between each entry is enough.

Caution: Keep in mind that manual modifications are lost after the installation of an Administration UI patch.

It is also important to remember that the higher the timeout, the higher the memory consumption will be.

After such a modification, make sure to restart the Administration UI and to inform the users about the downtime.

Configuration

Make sure to back up the files described in the following sections before modifying them.

To configure the Administration UI session timeout, modify the following:

`userauth-filter.sessionTimeOut` in the file `/opt/OV/OMU/adminUI/conf/auth.properties`.

The value is in seconds.

The default value configured is 900 seconds (15 minutes).

Note: The <session-timeout> (minutes) in the file: `/opt/OV/OMU/adminUI/webapps/midas/work/webapp/WEB-INF/web.xml` is the generic Webapp Session Timeout. A maximum value of 10 hours is assigned. This value should not be changed manually.

Enabling the Advanced Raw Editor

In addition to offering the standard raw editor capabilities, the Administration UI enables you to edit policies with a large number of conditions (over 700) in the raw mode and to edit text in the full screen mode by using the advanced raw editor.

By default, the standard raw editor is used. To enable the advanced raw editor, set the `OPC_USE_ADVANCED_RAW_EDITOR` configuration variable to `TRUE` as follows:

```
ovconfchg -ovrg server –ns adminui –set OPC_USE_ADVANCED_RAW_EDITOR TRUE
```

For more information about server configuration variables, see the *HPOM Server Configuration Variables* document.

Customizing the Administration UI

This section describes how to customize the Administration UI after installing HPOM (and consequently the Administration UI).

Tuning Java Parameters

The information in this section describes how to set up important Java parameters (for example, the amount of virtual memory that is available or the size of the stack for Java threads) and how to control

server start-up times. The information covers the following areas:

- "Virtual Memory " on page 480
- "JRE Startup " on page 481

Virtual Memory

For large numbers of users or objects, and if the system the Administration UI server is running on has plenty of RAM installed, performance can be improved significantly by allowing the JRE to obtain more virtual memory.

The recommended maximum amount of RAM is 1024 MB or 2048 MB where there is enough physical memory available.

Note: After the maximum heap size is increased to the range of 1500MB through 3500MB on HP-UX 11i v2 or higher, running the `java` command automatically launches the `java_q4p` executable instead of `java`. To avoid the false `Aborted` message for the Administration UI from appearing in the output of the `ovc -status` command, follow these steps:

1. Edit the `/var/opt/OV/conf/ctrl/adminui.xml` file by changing the line where the string value of the `ProcessDescription` element is defined. Replace `java` with `java_q4p`:

   ```
   <ovc:ProcessDescription>java_q4p</ovc:ProcessDescription>
   ```

2. Restart the `ovcd` process as follows:

   ```
   ovc -kill
   ovc -start
   ```

To change the memory setting, follow these steps:

1. In the `/opt/OV/OMU/adminUI/conf/servicemix/wrapper.conf` file search for the following block:

   ```
   [...]
   # Maximum Java Heap Size (in MB)
   wrapper.java.maxmemory=512
   ```

   When you find the needed block, change it to the following:

   ```
   Maximum Java Heap Size (in MB)
   wrapper.java.maxmemory=1024
   ```

2. Restart the application by running the following command:

   ```
   /opt/OV/OMU/adminUI/adminui restart
   ```

Do not decrease the value below the initial setting because by doing so you decrease the performance as well, which may result in the Administration UI not functioning properly.

Caution: Make sure to inform the users about the downtime.

JRE Startup

On slow systems, the startup of the Administration UI server may take some time. To prevent the controlling wrapper process from restarting the JRE, the following parameter may be increased (in bold):

```
vi /opt/OV/OMU/adminUI/conf/servicemix/wrapper.conf
```

```
[...]
# Number of seconds to allow for the JVM to be launched and contact the
# wrapper before the wrapper should assume that the JVM is hung and
# terminate the JVM process. Administration UI means never time out.
# Defaults to 30 seconds.
wrapper.startup.timeout=300
# Number of seconds to allow between the wrapper pinging the JVM and
# the response. Administration UI means never time out. Defaults to 30 seconds.
wrapper.ping.timeout=100
# Number of seconds to allow for the JVM to shutdown before the wrapper # should
assume that the JVM is hung and terminate the JVM process.
# Administration UI means never time out. Defaults to 30 seconds.
wrapper.shutdown.timeout=300
[...]
```

After any of the changes, restart the application by running the following command:

```
/opt/OV/OMU/adminUI/adminui restart
```

Caution: Make sure to inform the users about the downtime.

Replacing Wrapper Binaries Inside a Solaris Zone

It is highly recommended to replace the existing bundled wrapper binaries inside a Solaris zone with the latest ones. To do so, follow these steps:

1. Download the matching Community binaries package for your Solaris SPARC system from the following location:

   http://wrapper.tanukisoftware.org/doc/english/
2. Extract the following binaries from this package:

   `bin/wrapper`

   `lib/wrapper.jar`

   `lib/libwrapper.so`
3. Copy the `wrapper` file to the `/opt/OV/OMU/adminUI/` directory and the `wrapper.jar` and `libwrapper.so` to the `/opt/OV/OMU/adminUI/bin` directory.
4. Restart the Administration UI by running the following commands:

```
/opt/OV/OMU/adminUI/adminui clean
/opt/OV/OMU/adminUI/adminui start
```

SSH-based Agent Installation

This section describes the SSH-based installation of the HP Operations Agent by using the Administration UI.

In this section, you can find detailed information about the following topics:

Installation Overview

HPOM provides you with different methods for installing the agent software on a managed node. The agent installation can be triggered from the Administration UI or by using the command line interface. In both cases, the `inst.sh` script is used.

You can perform a clean installation of the agent or update the existing installation (for example, by using an agent patch).

Keep in mind that the Administration UI is a web-based GUI and it is not easily possible to start `inst.sh` as a child process and display input/output in a terminal.

Thus, the Administration UI agent installation is purely non-interactive and works in a request/response fashion. All parameters are collected, checked, and then passed on to the `inst.sh` script. When the `inst.sh` script finishes, the results are sent back to the Administration UI and saved in an agent installation log file.

SSH-based Agent Installation

If you want to use SSH during the agent installation, make sure that you select the Use SSH during installation check box. To do so, select Edit in the Action menu of the node. Inside the Installation tab, select the Use SSH during installation check box.

The following prerequisites must be met:

- Generate SSH keys on the HP Operations management server
- For each node, connect manually to the node and accept the host key.
- Copy the public key of the HP Operations management server to the target node.
- For each node, check if a connection can be established (without being asked for a password).
- Edit `/opt/OV/OMU/adminUI/conf/ovoinstall.properties` by commenting out the following line:

  ```
  ovoinstall.sshCmd = ssh -o StrictHostKeyChecking=no
  ```
- Restart the Administration UI by running the following command:

  ```
  ovc -restart adminui
  ```
- Do a test installation by using the standard mode (for example, `remsh`).

Pre-installation Options

Make sure the installation options are set correctly before you start with the agent installation. The default configuration values are sufficient if you use OpenSSHclient (or `ssh` client, which is similar to OpenSSH) for remote login. However, if you use non-OpenSSH client, make sure you have the additional settings defined. Consider the following options provided in the installation configuration file:

- `useOpenSSH` - This property indicates that used SSH client is similar to OpenSSH client. The default value is `TRUE`. Change this property in case there are some blocked process with non-OpenSSH client.

  If you change the default value to `ovoinstall.useOpenSSH=false`, make sure that the `sshCmd` property does not contain options such as: `sshCmd= ssh -o StrictHostKeyChecking=no`. If this is a case, leave these options commented or specify the following: `sshCmd=ssh`.
- `sshOptions` - This property is read-only if `useOpenSSH` is not set or if it is set on `FALSE`. Options must be separated with the comma ",". For example:

  ```
  ovoinstall.useOpenSSH=false
  ovoinstall.sshOptions=StrictHostKeyChecking=no,BatchMode=yes.
  ```

To see which command is run, check log files located at `/opt/OV/OMU/adminUI/logs/agent/install/jobs`.

Agent Installation Start

The agent installation can be started from the following two locations:

- For multiple installations or deinstallations: from the Administrative menu, select **Deployment > Agent (De)Installation...**.

  The next step is to choose the installation mode (that is, Installation or De-Installation).
- Single context-based installation: use the context-based Action menu of a node and select **Install**

Agent.

After you choose the installation mode, press the ... (Browse) button, and then select the set of target nodes from the Selector window.

The node names can also be typed into the text field. In this case, the name must be identical to the primary hostname as configured in HPOM (there is no name resolution).

Select as many nodes as needed, and then click **Close** to close the Selector window. After that, click the **Preinstall check** to start the prerequisite analysis. This analysis validates all selected nodes.

Preinstall Check Result

When the prerequisite analysis is complete, the list of target nodes is displayed. You must check the analysis result for each node. The important columns are the following:

- **Status**

  The following statuses are possible:

  - **ready**

    All prerequisites are met and the installation can proceed.

  - **invalid**

    Because one or more errors occurred, the installation is not possible.

  - **passwdrequired**

    All prerequisites are met, but the installation method requires a password that must be entered manually.

- **Method**

  For details, see "Installation Method" on page 485.

- **Comment**

  Lists the reason for each status result. For more information, click ?.

The nodes that are considered as installable are automatically selected. If you want to exclude individual nodes from the actual installation, clear the **Select** check box on the left of the node name. The nodes where an error occurred during the prerequisite check are automatically cleared and cannot be selected.

If required, enter a password. To reinstall an existing agent, select the **Force** check box.

Caution: Only the nodes that pass the analysis phase can be installed.

If no nodes are found as installable, the Install ... button is grayed out and disabled.

Installation Method

When the prerequisites analysis is finished, the resulting installation method is shown. For details, see Figure 19. This figure shows under which circumstances the corresponding method is determined.

The standard method represents the classic FTP/rexec communication that was not verified. It remains as the only possible method after all other options turned out as inappropriate. This standard method may also involve a rhosts-based RCP.

Caution: The status and method values do not guarantee that the installation will succeed but rather represent a best-effort prerequisites check to identify typical problems at the very beginning.

The following methods are possible:

- **local**

  Applies to the local HP Operations management server only.
- **HTTPS**

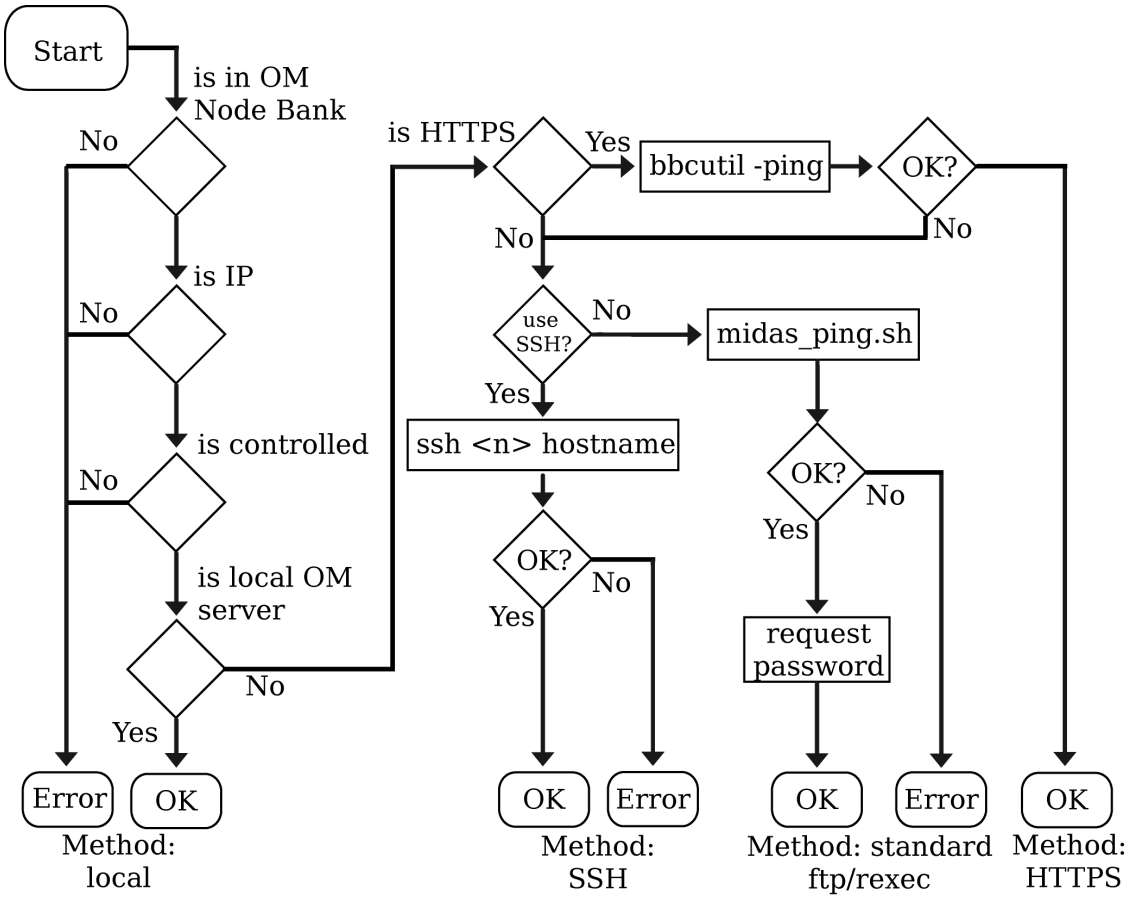
  A remote HP Operations Agent could be contacted and the HPOM built-in HTTPS communication will be used.
- **SSH**

  The SSH installation is configured for this node.
- **standard**

  The classic `FTP/rexec` communication will be attempted.
- **invalid**

  An error occurred (check the comment column).

Figure 19: Preinstallation Analysis Flow Chart



States and Error Codes

The following is a list possible errors:

- **nosuchnode**

  The node is not found at all or on the IP.
- **notcontrolled**

  The node is a member of the HPOM Node Bank, but it is not set as `controlled`.
- **sshnopubkey**

  The SSH installation was requested, but key-based SSH failed (an exit code of non-zero).
- **pingfailed**

  The node is not reachable through ICMP.

For more details, click **?**.

Note: passwdrequired is not an error. It means that the FTP/rexec method detected and the password is required.

Details about the Preinstallation Analysis

The flow chart shown in Figure 19 describes the steps performed during the prerequisite analysis for each node.

The commands are run as subprocesses of the Administration UI BackEnd server on the HP Operations management server. The output of the command can be viewed in the results list of the prerequisite analysis.

The SSH and ICMP `ping` commands are partially configurable using the Administration UI property file:

`/opt/OV/OMU/adminUI/conf/ovoinstall.properties`

These tests are the same tests that are also performed by the `inst.sh` script before starting the agent installation.

Main Installation Phase

After clicking the Install ... button, the Administration UI generates a parameter file for each selected node and runs the `inst.sh` script with these parameter files. The actual installation may take a long time. Therefore, the Administration UI performs this operation asynchronously in the background to avoid communication timeouts.

The output of the `inst.sh` script is stored in the Administration UI BackEnd module (on the HP Operations management server) and can be reviewed later by following the appropriate link.

Caution: The installation occurs in a sequence (one node after another because the `inst.sh` script cannot run concurrently) and may take a long time. Therefore, the installation logs appear one after another. Refresh the log file view from time to time.

Installation Log

The content of the installation log is written by the `inst.sh` script. View this output to see if the actual agent installation was successful or not.

In the Administration UI overview page, an exit code of zero is considered as successful.

SSH Details

Unlike FTP or RCP/REXEC, SSH is considered as a secure communication method.

SSH access is performed in a non-interactive way. This means that the public-key based access to the target node must be enabled, as it is not possible to provide a password to the SSH command. Keep in mind the following:

- The host key of the target node must be present in the SSH `known_hosts` file locally (that is, on the HP Operations management server). To accomplish this, choose one of the following:
  - Manually run the SSH command and confirm the host key.
  - Configure the SSH command to automatically accept new host keys. This configuration can be done globally, per user (`root`), or for the Administration UI only.
- The public key of the local `root` user (on the HP Operations management server) must be added to the SSH `authorized_keys` file of the installation user configured for that node (usually `root`) on the target node.
- A key may have a passphrase. If this is the case, it must be registered with SSH-add to the environment of the Administration UI server.
- An automatic initial host key acceptance is possible.

The local key of the `root` user on the HP Operations management server may be protected by a passphrase. To make this passphrase available to the Administration UI BackEnd server process, which starts the SSH command as a subprocess, use the `ssh-add` command to register the passphrase with `ssh-agent`. The Administration UI BackEnd server process must inherit some environment variables pointing it to `ssh-agent`. This can be accomplished by starting the `ssh-agent` process during the operating system boot (or any time before starting the Administration UI BackEnd server) and writing the printed environment variables into the `ssh.env` file.

For example:

```
ssh-agent -s > /opt/OV/OMU/adminUI/ssh.env
cat /opt/OV/OMU/adminUI/ssh.env
SSH_AUTH_SOCK=/tmp/ssh-DJBhmN5478/agent.5478; export SSH_AUTH_SOCK;
SSH_AGENT_PID=5479; export SSH_AGENT_PID;
```

After that, send the passphrase to ssh-agent:

```
ssh-add
Enter passphrase for /.root/.ssh/id_dsa:
Identity added: /.root/.ssh/id_dsa (/.root/.ssh/id_dsa)
```

In case of problems, see "Troubleshooting SSH" on page 488.

Troubleshooting SSH

The SSH key must be stored locally in the `known_hosts` file. Make sure the hostname you use in SSH commands is the same hostname the key is associated with. You must also keep in mind that short and long hostnames may not be recognized as the same.

The first time you try to connect through SSH to a particular hostname, the confirmation is required.

Because the Administration UI runs non-interactively, no confirmation of the host key is possible. Therefore, the host key confirmation must be accomplished in one of the following two ways:

- By manually running the `ssh root@targetnode` command and confirming the remote host key. This must be done before attempting the Administration UI-based agent installation. Keep in mind that different hostnames representing the same node are not considered as identical by the SSH command.
- By configuring the Administration UI to automatically accept remote host keys (that is, by specifying the `StrictHostKeyChecking=no` option).

Locking

The regular Administration UI locking mechanism is implemented to prevent multiple users from opening the agent installation view at the same time. In addition, the `inst.sh` script maintains a lock file that guarantees that only one instance of `inst.sh` runs at a time.

The HPOM lock is removed by `inst.sh` unless the `inst.sh` script was killed with SIGKILL (signal 9) or crashed. In this case, the lock file remains and subsequent executions of `inst.sh` fail because of the wrong assumption that `inst.sh` is still running. To address this issue, you can choose between the following options:

- Cancel

  Enables you to cancel the operation and retry later.
- Refresh

  Enables you to retry testing the HPOM lock file (for example, if the `inst.sh` script was started 5 minutes ago).
- Ignore

  Enables you to remove the lock and continue. This option may be appropriate if the lock is older than a certain time (for example, one hour) and/or no `inst.sh` process is running anymore.

Configuration and Tuning

To adapt the behavior of SSH and ICMP `ping` run during the prerequisite analysis phase, configure the properties as follows:

/opt/OV/OMU/adminUI/conf/ovoinstall.properties

```
[...]
Local SSH command to check SSH access in pre-install phase:
ovoinstall.sshCmd=ssh

Command executed remotely by SSH:
ovoinstall.sshCmdRem=hostname
```

```
Local command to ping node in pre-install phase:
ovoinstall.pingCmd=midas_ping.sh
```

The `sshCmd` property can be used to configure the exact command run to test SSH connectivity. For example, a full path may be specified or additional options. The latter may be necessary to enable automatic host key acceptance. This can be accomplished by configuring:

```
ovoinstall.sshCmd = ssh -o StrictHostKeyChecking=no
```

The `sshCmdRem` property can be used to change the command run remotely. By default, this is the hostname, but any other remote command can be configured. The exit code of zero must be returned.

During the run time, the SSH command is as follows:

<sshCmd> <debug options> `-l root -o BatchMode=yes` *<target node> <sshCmdRem>*

In this instance, *<sshCmd>* and *<sshCmdRem>* are substituted by the properties specified above, *<debug options>* is either empty or `-v -v -v` in the `DEBUG` mode, and *<target node>* is the primary hostname (as configured in HPOM) of the target node.

To enable ICMP, remove `#` before the `ovoinstall.pingCmd=midas_ping.sh` line. The ICMP `ping` behavior can be customized in the following file:

```
/opt/OV/OMU/adminUI/bin/midas_ping.sh
```

The `midas_ping.sh` script is a wrapper script that calls the actual `ping` command depending on the architecture of the target node. If you want no `ping` at all, you can edit `midas_ping.sh` so that it directly returns 0 instead of doing anything.

Troubleshooting

In the prerequisites analysis phase, do as follows:

- Review the output of a precheck command (click **?**).
- Set the Administration UI to the DEBUG mode (especially for SSH).

During the agent installation phase, do as follows:

- Review the generated parameter file.
- Review the installation log file (the output of `inst.sh`).
- Run the `inst.sh` script manually.

Note: The steps differ depending on the installation phase.

The installation always invokes the `inst.sh` script. If the prerequisite analysis succeeds, a parameter file is generated and passed on to the `inst.sh` script. If the installation fails, the `inst.sh` script prints the diagnostic output that can be viewed in the installation log files.

Setting the Administration UI to the DEBUG mode can be done by editing the following file:

```
/opt/OV/OMU/adminUI/conf/log4j.xml
```

Change the log level to `DEBUG` in the following section:

```
<logger name="com.bes.ovo.comp.install" additivity="false">
 <level value="DEBUG"/>
 <appender-ref ref="ovo"/>
</logger>
```

No restart of the Administration UI software is necessary. A short time after the modification, `DEBUG` logging is enabled. After testing, review the following log file:

```
/opt/OV/OMU/adminUI/log/ovo.log
```

Particularly for SSH-based nodes, the SSH command run during the preinstall phase, will run in the verbose mode and generate a detailed diagnostic output.

Troubleshooting Administration UI

This chapter describes some of the problems that might occur and explains how you can use the available tools to access the information you need to start the process of fixing them.

In this chapter, you can find the information about the following topics:

General Procedures

When you encounter problems, follow these guidelines:

- Describe the problem as precise as possible:
  - Error reports saying only that you cannot edit a policy are not helpful.
  - Provide screenshots, shell output data, log files, or the policy.
  - Provide the `support.zip` file.
- Think about differences:

- Differences to other instances (“It works for all items, except this”). What is special about the non-working item?
- “Yesterday it used to work”. So, what has changed since then?

- Try to determine if the problem is within HPOM or the Administration UI:
  - Can you perform the same operation using native HPOM tools (for example, `opcragt`, `opchbp`, and so on)?
  - Are there any related entries in the HPOM error log files?
  - Use HPOM tracing. Most Administration UI operations are performed by calling the HPOM API. This can be traced using the regular HPOM tracing facility. The trace area of most interest is `opc.api` (HPOM 8 XPL tracing).

    You may have to restart the Administration UI application to be traced.
  - Check the HPOM database where most of the management server information is stored. You can use the `sqlplus` or `psql` tool to check the internal data. However, it is not recommended to modify database data directly, as this might lead to unexpected problems or data corruption.
- Verify operating system resources.

  Insufficient disk space or full kernel tables are common causes. Such problems may also be logged in `syslog` (Unix).

In addition, most efficient troubleshooting strategies involve the following:

- Performing simple steps to include or exclude the most likely causes:
  - Running commands that provide fast results (for example, `adminui clean` and `adminui analyze`)
  - Checking error log files
- Tracing (for example, configuring the trace level, restarting services, reviewing trace data, and so on)
- Determining (for example, also determining that something is not the problem may help).

Note: Make sure that you always provide the support team with the `support.zip` file (see "Packing up Support Data " on page 495).

Display-related Problems

Because the Administration UI is a web-based tool that uses a browser to connect to a server, it is unlikely that you will encounter problems displaying the GUI. However, there are some exceptions, notably on UNIX operating systems, where you might need to investigate further. For example:

- You are trying to redirect the display of the web browser between UNIX hosts.
- You are installing the Administration UI software on a UNIX host.

In these two cases, check the following:

- `xhost` settings:

  Allow remote X access on the host where the GUI is supposed to appear. To configure X access on a UNIX host, use the `xhost +` command.

- `DISPLAY` variable:

  Set the `DISPLAY` variable on the UNIX host where the program that starts the display is running.

  Use the `export DISPLAY=<hostname>:0` command to set the display.

If the display settings are correctly set and the problem persists, check the following known problems:

- LANG environment variable:

  A missing or incorrectly set language variable occasionally produces the following (or similar) error message when running a command on HP-UX systems:

  ```
  Warning: Missing charsets in String to FontSet conversion
  Warning: Unable to load any usable fontset
  ```

  If the language environment variable is not set or is set incorrectly, set the language variable as part of the command:

  ```
  LANG=C.iso88591 <the failing command>
  ```

Menu Display Problem

If there is a problem with a menu display, it might be caused by the cache of the web browser. In this case, force a cache refresh in the browser.

Cache refresh command reloads the page without using the existing cached data. The menus are displayed horizontically instead of vertically.

After a successful logon, sometimes the `Help page not found` error message appears. The reason is a browser bookmark containing a URL from a previous Administration UI version. For example:

- IPv4 address:

  ```
  http://192.168.10.88:9662/midas/<lang>/index.html
  ```

- IPv6 address:

  `http://[fec0::250:56ff:fea8:2ad2]:9662/midas/<lang>/index.html`

In these instances, `<lang>` is `en` for English or `ja` for Japanese.

Caution: With an IPv6 address, you can open the Administration UI by using a browser only on systems where IPv6 is enabled. Make sure you use the square brackets ([ ]) as shown in the example.

To prevent this from happening, shorten the URL to `http://address:port` and update the bookmark in your browser. For example:

- IPv4 address:

  `http://192.168.10.88:9662`
- IPv6 address:

  `http://[fec0::250:56ff:fea8:2ad2]:9662`

Using Log Files

The Administration UI writes detailed information about run-time operations to a number of different platform-specific and adapter-specific log files. The name of the log file and the information it contains varies according to the adapter writing the log file. By default, the Administration UI stores server log files in the following location:

`/opt/OV/OMU/adminUI/logs/*.log`

The Administration UI WebApp component writes information to its own log files that are stored in the following location:

`/opt/OV/OMU/adminUI/webapps/<comp>/work/webapp/WEB-INF/logs/*.log`

In this instance, the default path to the log files written by the Administration UI Web Application, <*comp*>, stands for one of the following values:

- `midas`

  Name of the logical Administration UI WebApp component

By default, the Administration UI logs basic information about errors that occur during normal operation. However, if you are troubleshooting a particular problem and require more information, you can increase the log verbosity level. This can be done for each individual component for which you require more information. For information about the trace levels that are allowed and instructions on how to set the trace level, see "Logging and Tracing Mechanism" on page 453.

For a list of all main log files, see "Log Directory" on page 409. Some log files belong to a specific adapter. If you are troubleshooting a problem related to a specific adapter, check the log files that the

adapter writes to. For example, if you are investigating a problem related to a certain HPOM area (for example, listing all policies does not work, and so on), check the `ovo.log` file.

Note: When analyzing start-up problems, the suggested route is:
`wrapper.log -> servicemix.log -> ovo.log -> midas.log`

Viewing Raw XML Data

In case of problems displaying data correctly in the web browser, raw XML data can be displayed. Displaying raw XML data can help you to determine if there is a problem with actual data or a failure in the presentation layer of the Administration UI.

To view raw XML data, follow these steps:

1. Modify the standard URL by changing the `.../*-INC-/...` string to **`.../*-RAW-/...`**
2. At the end of the URL, add the following: **`&cocoon-view=raw`**

Note: If the graphical web interface is not returning a proper result and web page this would indicate that perhaps there is a problem with one of the stylesheets which are used to render the web interface or result output.

Troubleshooting Commands

This section describes advanced options for the `adminui` command.

Note: You do not need these advanced options for everyday maintenance. However, in troubleshooting situations, these commands might be helpful. Check with Product Support before using them.

The information in this section covers the following topics:

- "Packing up Support Data " on page 495
- "Clean Restart " on page 496
- " Checking Component Status " on page 497
- "Reinitializing the XML Database " on page 498

Packing up Support Data

The `adminui support` command is an important support tool that you should use if you run into problems with the Administration UI and want to contact the Product Support team. With the `adminui support` command, it is possible to quickly collect all required log files and configuration files. The file location is inside the following directory:

`/opt/OV/OMU/adminUI/`

Example:

```
/opt/OV/OMU/adminUI/adminui support
```

The shell output looks as follows:

```
[root@deli:/opt/OV/OMU/adminui] ./adminui support
[...]
support.zip:
     [echo] collecting support information ...
     [echo] collecting version info ...
     [echo] collecting installed files ...
     [echo] collecting Java properties ...
[propertyfile] Creating new property file:
[...]
intern.checksum_check:
     [echo] checking checksums ...
     [echo] creating support zip ...
      [zip] Building zip: /opt/OV/OMU/adminUI/support_20090325162209.zip
[echo] cleaning up ...
[echo] send the file /opt/OV/OMU/adminUI/support_20090325162209.zip to support
BUILD SUCCESSFUL
Total time: 2 minutes 30 seconds
```

At the end of the output you see the support file name and the location.

The file name contains the date and time when the command was run.

The zip file contains the following:

- All core configuration files from `/opt/OV/OMU/adminUI/conf`
- All core log files from `/opt/OV/OMU/adminUI/logs`
- All WebApp component log and configuration files from `/opt/OV/OMU/adminUI/webapp/midas/work/webapp/WEB-INF`
- Some environment variables (for example, output from `adminui analyze`, `uname -a`, `listener.ora`, and so on)

Clean Restart

The `adminui clean` command is helpful in solving any existing problems, especially when a file corruption exists that prevent one or more modules to start up successfully. In this case, perform a clean restart of the application by running the following commands:

```
/opt/OV/OMU/adminUI/adminui clean
```

```
/opt/OV/OMU/adminUI/adminui start
```

These commands restart the application performing a cleanup of all log files and run-time files, forcing the application to unpack all necessary run-time files again.

Caution: Make sure no other users are logged on before you restart the application. Otherwise, their current work might be lost.

Checking Component Status

The `adminui servicemix` command displays information about installed Servicemix JBI components (binding components and services) and run-time information about deployed service assemblies and their contained service units.

With this command you can check if all required Administration UI adaptors were successfully started (missing service assemblies indicate a problem with the corresponding adapter). In this case, review the `servicemix.log` and `midas.log` files.

The `adminui servicemix` command is not needed during day-to-day operation but can be used for extended troubleshooting purposes. It is recommended to redirect the output to a temporary file:

```
/opt/OV/OMU/adminUI/adminui servicemix > /tmp/sm.status
```

The following example shows parts of the output of the `adminui` servicemix command:

```
servicemix:
list-binding-components:
[echo]  list-binding-components
[echo]  Prints information about the binding components installed in servicemix.
[echo]     host=avocado
[echo]     port=9660
[...]
list-service-engines:
[echo]  list-service-engines
[echo]  Prints information about all of the Service Engines in Servicemix.
[echo]     host=avocado
[echo]     port=9660
[...]
[jbi:list-service-engines]<component-info type='service-engine' name='servicemix-
eip' state='Started'>
[jbi:list-service-engines]   <component-info type='service-engine'
name='servicemix-lwcontainer' state='Started'>
[jbi:list-service-engines]   <component-info type='service-engine'
name='servicemix-script' state='Started'>
[...]
list-service-assemblies:
[echo]  list-service-assemblies
[echo]  list deployed Service Assemblies in Servicemix.
[echo]     host=avocado
[echo]     port=9660
[...]
[jbi:list-service-assemblies]<service-unit-info name= 'midas-ovoconfig'
state='Started' deployed-on='servicemix-lwcontainer'>
[jbi:list-service-assemblies]<description>MIDAS HPOM for UNIX Configuration
```

```
Adaptor</description>
[...]
```

The `list-binding-components` and `list-service-engines` blocks must be always present including some additional details.

The `list-service-assemblies` block shows all deployed service assemblies. This list must match the set of files in `/opt/OV/OMU/adminUI/deploy`. If a service assembly is present in the `deploy` directory but not listed in the output of the `adminui servicemix` as `Started`, it failed to start. In this case, view the log files for details.

For example, the `servicemix.log` file can contain entries similar to the following:

```
ERROR - 2009-02-20 13:03:16,369 | AutoDeploymentService.updateArchive(308) | Failed
to update Service Assembly: midas-wapam
java.lang.Exception: <?xml version="1.0" encoding="UTF-8"?>
[...]
nested exception is java.lang.UnsatisfiedLinkError: no jpam in
java.library.path</loc-message>
[...]
```

In this example, the `midas-wapam` adapter (the PAM authentication adapter) failed to start because the `libjpam.so` native library could not be found.

Reinitializing the XML Database

The `adminui init` command clears and reinitializes the XML database.

For details, see "Reinitializing the XML Database" on page 441.

Communication Problems

This section describes how to investigate problems relating to inter-component communication. For example, you can learn how to perform basic checks to ensure that name resolution works correctly or to ensure that advanced features work as expected.

The information in this section covers the following areas:

General Communication Problems

Make sure that general network connectivity exists between the involved systems. Because the Administration UI is installed on the HPOM system, end users must be able to reach and access the HPOM system from their workstations using their web browsers on the correct Web Application UI ports (the default is 9662 for `http://` and 9663 for `https://` requests).

It is common that the end user is in a different network than the HP Operations management server. If a firewall exists, the necessary ports of the Administration UI WebApp component must be open. In addition, the hostname of the HPOM system must be resolvable within the end users network.

Make sure that you check the following:

- Check whether name resolution works correctly, type:

  `nslookup <target-node>`

  Make sure this returns the same IP address on both systems (unless there is a NAT router).

- Ping the target system, type:

  `ping <target-node>`

- Check if the Administration UI WebApp port can be reached:

  `telnet <target-node> <port>`

  For example:

  ```
  telnet ios 9662

  Trying 192.168.123.113...
  Connected to ios.
  Escape character is '^]'.
  ^]
  telnet> quit
  Connection closed.
  ```

No real communication is possible, but the `telnet` command should at least establish a connection.

PAM Integration

For the PAM setup, see "PAM Authentication" on page 507.

If you encounter problems with a configured PAM module, follow these guidelines:

- Check if there are any related entries in the following log file:

  `/opt/OV/OMU/adminUI/logs/usermgmt.log`

  For example:

```
DEBUG - 2006-12-20 14:03:49,002 | UserModelRequestTransformer.transform(?)
|rewriting request to service pam
DEBUG - 2006-12-20 14:03:49,087 | PamServer.authenticate(?) | Authenticating user
admin with PAM service midas ...
DEBUG - 2006-12-20 14:03:49,088 | Pam.authenticate(160)| Debug mode active.
ERROR - 2006-12-20 14:03:51,126 | PamServer.authenticate(?) | Authentication of
user admin failed: Underlying authentication service can not retrieve
authentication information.
DEBUG - 2006-12-20 14:03:51,163 | UserMgmtFilter.onMessageExchange(?) | clearing
response { DOType: errorresponse
extra : com.bes.itm.comp.usermgmt.AuthenticationFailedException: Could not
authenticate user admin via PAM. PAM error: Underlying authentication service can
not retrieve authentication information.
```

- Enable additional tracing in the PAM adapter module. Configure the following in `/opt/OV/OMU/adminUI/conf/log4j.xml`:

  ```
  <logger name="net.sf.jpam" additivity="false">
   <level value="DEBUG"/>
   <appender-ref ref="usermgmt"/>
  </logger>
  ```

  The PAM module writes additional debug statements into the `usermgmt.log` file.

- Perform a stand-alone test of the authentication method:

  ```
  cd /opt/OV/OMU/adminUI/
  ```

  ```
  export SHLIB_PATH=$SHLIB_PATH:./lib/midas
  ```

  ```
  ./jre/bin/java -cp ./lib/cli/midas_cli.jar:./work/service-assemblies/midas-
  wapam/version_1/sus/servicemix-lwcontainer/midas-pam/lib/jpam-
  1.1.jar:./lib/commons-logging-1.1.jar com/bes/itm/comp/usermgmt/TestPam <user
  name> <password>
  ```

  ```
  **** Starting ...
  **** Authenticating user admin ...
  **** Authentication done.
  **** Success: false
  **** Result: Underlying authentication service can not retrieve authentication
  information.
  **** Exit.
  ```

  This test class performs pure PAM authentication of <*user name*> with <*password*> and prints the results to `stdout`.

- Check the PAM configuration as described in "PAM Authentication" on page 507.

- Make sure that all dependencies of the native library `/opt/OV/OMU/adminUI/lib/midas/libjpam.so` are met.

  For example:

  ```
      ldd < _HOME>/lib/midas/libjpam.so
    linux-gate.so.1 =>  (0xffffe000)
  ```

```
libpam.so.0 => /lib/libpam.so.0 (0x40016000)
libpam_misc.so.0 => /lib/libpam_misc.so.0 (0x40020000)
libdl.so.2 => /lib/libdl.so.2 (0x40023000)
libc.so.6 => /lib/tls/libc.so.6 (0x40027000)

/lib/ld-linux.so.2 (0x80000000)
```

- Review the UNIX `syslog` log file (the native PAM library logs messages to `syslog`).

  For example:

  ```
  grep -i pam /var/log/messages
  ```

  ```
  [...]
  Dec  1 18:15:41 garlic midas.pam(pam_unix)[25305]: authentication failure;
  logname= uid=0 euid=0 tty= ruser= rhost=  user=admin
  Dec  1 18:17:40 garlic midas.pam(pam_unix)[25305]: authentication failure;
  logname= uid=0 euid=0 tty= ruser= rhost=  user=admin
  [...]
  ```

- Enable tracing the native PAM library, turn on debugging on the `syslog` level. The related service name is `auth`. To turn on tracing, configure `syslogd` as in the following example:

  ```
  touch /etc/pam_debug
  ```

  ```
  vi /etc/syslog.conf
  ```

  ```
  auth.debug              /tmp/pam_auth.log
  ```

  ```
  [...]
  ```

  To reload the configuration of the `syslogd` process, run the following command:

  ```
  kill -HUP `cat /var/run/syslog.pid`
  ```

  The resulting debug output looks as in the following example:

  ```
  tail -f /tmp/pam_auth.log
  ```

  ```
  Dec 20 15:41:16 ios PAM: pam_start(midas admin)
  Dec 20 15:41:16 ios PAM: pam_set_item(1)
  Dec 20 15:41:16 ios PAM: pam_set_item(2)
  Dec 20 15:41:16 ios PAM: pam_set_item(5)
  Dec 20 15:41:16 ios PAM: pam_set_item(6)
  Dec 20 15:41:16 ios PAM: pam_authenticate()
  Dec 20 15:41:16 ios PAM: load_modules: /usr/lib/security/hpux32/libpam_unix.so.1
  Dec 20 15:41:16 ios PAM: load_function: successful load of pam_sm_authenticate
  Dec 20 15:41:16 ios PAM: pam_get_username(ux)
  Dec 20 15:41:16 ios PAM: pam_mapping_in_use()
  Dec 20 15:41:16 ios PAM: pam_set_item(6)
  Dec 20 15:41:16 ios PAM: pam_acct_mgmt()
  Dec 20 15:41:16 ios PAM: load_modules: /usr/lib/security/hpux32/libpam_unix.so.1
  Dec 20 15:41:16 ios PAM: load_function: successful load of pam_sm_acct_mgmt
  Dec 20 15:41:16 ios PAM: pam_get_username(ux)
  ```

```
Dec 20 15:41:16 ios PAM: pam_mapping_in_use()
Dec 20 15:41:16 ios PAM: pam_end(): status = Success
```

Direct LDAP Integration

For the LDAP setup, see "LDAP Authentication" on page 510.

If you encounter problems with a configured LDAP module, follow these guidelines:

- Check if there are any related entries in the following log file:

  `/opt/OV/OMU/adminUI/logs/usermgmt.log`

  For example:

  ```
  DEBUG - 2011-06-22 14:30:17,007 | UserModelRequestTransformer.onMessageExchange
  (?) | rewriting request to service ldap
  INFO - 2011-06-22 14:30:17,036 | UserMgmtFilter.providerModeXML(?) |
  UserMgmtFilter->providerModeXML(): Loop over all requests and check role and
  attach filter and so on
  DEBUG - 2011-06-22 14:30:17,243 | LdapAuthenticationProvider.retrieveUser(142) |
  Retrieving user tge
  DEBUG - 2011-06-22 14:30:17,245 | DefaultInitialDirContextFactory.connect(215) |
  Creating InitialDirContext with environment
  {java.naming.provider.url=ldap://ldap-serv:389/dc=hp-intern,dc=com,
  java.naming.factory.initial=com.sun.jndi.ldap.LdapCtxFactory,
  java.naming.security.principal=cn=Manager,dc=hp-intern,dc=com,
  com.sun.jndi.ldap.connect.pool=true, java.naming.security.authentication=simple,
  java.naming.security.credentials=******}
  DEBUG - 2011-06-22 14:30:17,262 | FilterBasedLdapUserSearch.searchForUser(118) |
  Searching for user 'tge', in context
  javax.naming.directory.InitialDirContext@121eb03f, with user search [
  searchFilter: '(uid={0})', searchBase: 'ou=people', people: single-level,
  searchTimeLimit: 0 ]
  ...
  WARN - 2011-06-22 14:30:17,374 | UserModelServer.authenticate(?) | user
  authentication failed, credential is not valid
  ```

- Enable additional tracing in the LDAP adapter module. Configure the following in `/opt/OV/OMU/adminUI/conf/log4j.xml`:

  ```
  <logger name="org.acegisecurity" additivity="false">
   <level value="DEBUG"/>
   <appender-ref ref="usermgmt"/>
  </logger>
  ```

  The LDAP module writes additional debug statements into the `usermgmt.log` file.

- Perform a stand-alone test of LDAP.

For example:

```
cd /opt/OV/OMU/adminUI/
```

```
./jre/bin/java -cp

./lib/cli/midas_cli.jar:./work/service-assemblies/midas-belite/version_
1/sus/servicemix-lwcontainer/midas-auth/lib/acegi-security-1.0.0-
RC2.jar:./lib/spring-context-2.0.6.jar:./lib/spring-core-2.0.6.jar:./lib/spring-
beans-2.0.6.jar:./lib/commons-logging-1.1.jar com.bes.itm.comp.usermgmt.TestLdap
tge <passwd>

**** Starting ...
**** Loading ldap.properties file ...
**** Server URL: ldap://ldap-serv:389/dc=hp-intern,dc=com
**** Manager name: cn=manager,dc=hp-intern,dc=com
**** Manager passwd: trallala
**** Auth mode: BIND_WITH_DN
**** Search patterns: sn={0},ou=People
**** Group search base: null
**** Authenticating user tge ...
**** Failed to authenticate user:
org.acegisecurity.BadCredentialsException: Bad credentials
**** Exit.
```

This test class performs pure LDAP authentication of <*user name*> with <*password*> and prints the results to stdout. The LDAP configuration is determined from `./conf/ldap.properties`.

Checking the Process Status

If the JRE process that runs the Administration UI server crashes, collect the following data and send it to the Product Support team:

- `support.zip` file
- `hs_err_pid<PID>.log` file

If the crash occurs regularly, the core file written by the JRE may also help. Creating core files on UNIX is disabled by default. To enable the creation of core files, comment out the `ulimit -c 0` line in this script:

```
/opt/OV/OMU/adminUI/bin/server.sh
```

If the crash occurs again, save the core file for later evaluation. If a core file exists, it can be analyzed using the HPOM utility `stacktrace`, as shown in the following example (applies only to an HP-UX system):

```
/opt/OV/contrib/OpC/stacktrace /opt/midas31/core

/opt/midas31/jre/bin/IA64N/java
```

Authentication Problems

If you or another user cannot log on to the Administration UI web interface, check the following:

- Does the Administration UI run on the desired system and does it use the correct ports? Does the web interface show up?

  Check if the ports are allocated by using the `netstat` or `lsof` command.

  In addition, start a web browser locally on the WebApp system and try to connect.
- Can you reach the HPOM system from the basic network perspective?

  Use the `ping` and `telnet` commands.

Troubleshooting Tips

This section contains troubleshooting tips for the most common authentication problems that are currently known:

- Sometimes an error message appears after a new installation of the Administration UI, when you attempt to log on by entering the following URL:

  `http://<HPOM_management_server>:9662/`

  To solve this problem, follow these steps:

  a. Run the following command:

  ```
  /opt/OV/OMU/adminUI/adminui webassemblies
  ```

  This command stops the Administration UI and recompiles all web assemblies that are needed for a correct display of the Administration UI web interface. The following `BUILD SUCCESSFUL` message should appear:

  ```
  [war] Building war: /opt/OV/OMU/adminUI/work/tmp/webdeploy/midas.war
  [echo] copying war file to webapps
  [copy] Copying 1 file to /opt/OV/OMU/adminUI/webapps
  [delete] Deleting directory /opt/OV/OMU/adminUI/work/tmp/webdeploy
  [delete] Deleting directory /opt/OV/OMU/adminUI/webapps/midas
  [echo] done.  restart server.
  BUILD SUCCESSFUL
  Total time: 1 minute 53 seconds
  ```

  b. Restart the Administration UI by running the following command:

  ```
  /opt/OV/OMU/adminUI/adminui restart
  ```
- If a listing of the directories and files is displayed instead of the web interface, follow these steps:

  a. Run the following command:

  ```
  /opt/OV/OMU/adminUI/adminui webassemblies
  ```

  This command stops the Administration UI and recompiles all web assemblies that are needed for a correct display of the Administration UI web interface. The following `BUILD SUCCESSFUL` message should appear:

```
[war] Building war: /opt/OV/OMU/adminUI/work/tmp/webdeploy/midas.war
[echo] copying war file to webapps
[copy] Copying 1 file to /opt/OV/OMU/adminUI/webapps
[delete] Deleting directory /opt/OV/OMU/adminUI/work/tmp/webdeploy
[delete] Deleting directory /opt/OV/OMU/adminUI/webapps/midas
[echo] done.  restart server.
BUILD SUCCESSFUL
Total time: 1 minute 53 seconds
```

b. Restart the Administration UI by running the following command:

```
/opt/OV/OMU/adminUI/adminui restart
```

- If you cannot log on by using the default user name and password (that is, the interface informs you that the provided user name and password are incorrect), the reason might be one of the following:
  - If you have an expired HPOM license password, a warning message appears and the log-on fields are not shown.

    To solve this problem, follow these steps:

    i. Check the HPOM license password by running one of the following commands:

    ```
    ovolicense -s -p HPOM
    ovolicense -s -p HPOM | grep ovosv | grep -i critical;
    /opt/OV/bin/ovolicense -s -p HPOM | grep ovosv | grep -i locked
    ```

    ii. Install a valid HPOM license password.

    For example, to install a new license:

    ```
    JAVA_HOME=/opt/OV/nonOV/jre/b
    export JAVA_HOME
    /opt/OV/bin/ovolicense -gui -a HPOM
    /opt/OV/bin/ovolicense -install -category OMU -file lic.dat
    ```

    iii. After the license update, restart the Administration UI by running the following commands:

    ```
    /opt/OV/OMU/adminUI/adminui stop
    /opt/OV/OMU/adminUI/adminui clean
    /opt/OV/OMU/adminUI/adminui start
    ```

    Caution: If you still receive the warning message, delete your browser cache or refresh a web page.

  - Another possibility is that the user is in no user group or that this user group does not have a user role assigned inside the Administration UI.
    Solution: Try another user logon (for example, with the main Administration UI user `admin` to validate if the log-on process is generally broken or not).

  - Sometimes the user account is disabled, for example, because of multiple attempts to log on with

wrong credentials when the account locking is enabled.

Solution: Enable the account again as described in "Account Locking " on page 476.

- In case the previous tips do not help, a file corruption might exist, preventing some modules to start up successfully.
  Solution: Perform a clean restart of the application by running the following commands:

  ```
  /opt/OV/OMU/adminUI/adminui clean
  /opt/OV/OMU/adminUI/adminui start
  ```

  These commands restart the application performing a cleanup of all log files and run-time files, forcing the application to unpack all necessary run-time files again.

  Caution: Make sure no other users are logged on before you restart the application. Otherwise, their current work might be lost.

- If the XML database that stores the user database was not successfully initialized during the installation, all initial user data may be missing. Therefore, any log-on attempt fails.

  In this case, it is recommended to perform a complete reset of the XML database. The reset completely reinitializes the XML database.

  Note: The following command is recommended only after a new installation of the Administration UI:

  ```
  /opt/OV/OMU/adminUI/adminui init force
  ```

  Make sure that you do not perform the initialization before the Administration UI is started. Otherwise, the operation fails because there is no connection to the internal XML database.

  In this case, no restart of the Administration UI is necessary. The `BUILD SUCCESSFUL` message should appear and you should be able to log on with the default user name and password.

  Do not run this command later during normal operation unless you are advised to do so by Product Support because it will destroy and reinitialize all custom user configuration.

  For more information about the `adminui init force` command, see "Reinitializing the XML Database " on page 498.

- If a clean restart does not help, it might be possible that a rare case of corruption of the XML database exists.
  Solution: Create a backup of the Administration UI configuration including the XML database by running the following command:

  ```
  /opt/OV/OMU/adminUI/adminui save
  ```

In addition, create a support zip containing all Administration UI configuration and log files. Run the following command:

```
/opt/OV/OMU/adminUI/adminui support
```

Make sure that you send both zip file packages to the Product Support team explaining your problems so they can analyze the XML database files.

For detailed information about the `adminui save` and `adminui support` commands, see "Saving Configuration" on page 425 and "Collecting Support Information" on page 432.

External Software

This section lists additional external software products that are integrated in the Administration UI and describes how you can configure the software to suit the demands of your environment. All the software products described in this section are optional unless you choose to install and configure functionality on which an Administration UI feature depends.

The information in this section covers the following topics:

- "Authenticating Administration UI Users Using PAM or LDAP" on page 507
- "Daylight Saving Time (DST) Patches" on page 517

Authenticating Administration UI Users Using PAM or LDAP

This section describes how to authenticate Administration UI users using PAM or LDAP. Authentication of Administration UI users occurs inside the Administration UI WebApp server part to which the user's web browser connects.

Note: When setting up a new Administration UI user, make sure that the account exists in both the Administration UI and the external authentication system. In addition, the Administration UI user must be a member of at least one Administration UI group that has at least one Administration UI user role assigned.

PAM Authentication

To authenticate Administration UI users by using PAM, no extra software is needed because the Administration UI already includes the JPam open-source module. For details about JPam, see the following URL:

http://jpam.sourceforge.net

Note: PAM is an interface linking software that provides authentication services such as LDAP, Kerberos, and UNIX passwd to user applications such as the Administration UI. Therefore,

software modules that implement the actual authentication service may be required.

Caution: If you have the Administration UI integrated with Active Directory by using PAM authentication, logging on to the Administration UI may fail when a user password is about to expire.

To prevent this from happening, choose one of the following:

- Change the user password, and then revert it to the value before this change.
- Integrate the Administration UI with Active Directory by using the LDAP integration.

To configure PAM authentication, follow these steps:

1. Decide which authentication service to use. If needed, install required software modules and configure them.

   Caution: It is highly recommended that you perform a stand-alone test of the authentication service (that is, outside the Administration UI context).

2. Configure all Administration UI user accounts in the authentication service.
3. Set up PAM authentication on the HP Operations management server.
4. Configure PAM to send Administration UI authentication requests to the desired authentication service (the PAM service name is `midas`).

   Note: PAM configuration is platform dependent. For troubleshooting, contact your system administrator.

   For example, to use UNIX password authentication, perform the following:

   - On HP-UX:

     Edit the `/etc/pam.conf` file for the `midas` module by adding the following lines:

     ```
     midas auth required /usr/lib/security/hpux32/libpam_unix.so.1
     midas account required /usr/lib/security/hpux32/libpam_unix.so.1
     ```

   - On Solaris:

     Edit the `/etc/pam.conf` file for the `midas` module by adding the following lines:

     ```
     midas auth requisite pam_authtok_get.so.1
     midas auth required pam_unix_auth.so.1
     midas account required pam_unix_account.so.1
     ```

   - On RHEL:

Create the `/etc/pam.d/midas` PAM module, and then edit the `/etc/pam.d/midas` file by adding the following lines:

```
auth sufficient pam_unix.so nullok try_first_pass
auth required pam_deny.so
account required pam_unix.so
account required pam_permit.so
```

5. Activate the external authentication service in the `auth.properties` file by following these steps:
   a. Open the `auth.properties` file with the `vi` editor by running the following command:

      ```
      vi /opt/OV/OMU/adminUI/conf/auth.properties
      ```

   b. Edit the `auth.properties` file so that it contains the following:

      ```
      # external configuration file for complex authentication
      setups
      usermodel-router.authResource=file:conf/auth.xml
      # eof
      ```

6. Switch the Administration UI to PAM authentication by configuring the `auth.xml` file.

   The following is an example file:

   ```
   <?xml version="1.0" encoding="UTF-8"?>
   <!DOCTYPE beans PUBLIC "-//SPRING//DTD BEAN//EN"
   "http://www.springframework.org/dtd/spring-beans.dtd">
   <beans>
      <bean id="targetServices" class="java.util.ArrayList">
        <constructor-arg>
   <list>
     <value>pam</value>
     <value>usermgmt</value>
   </list>
        </constructor-arg>
      </bean>
   </beans>
   ```

   The Administration UI tries to use the PAM server for logon. If this authentication fails, the Administration UI tries standard “user management” authentication.

   If you want to set up only PAM authentication (that is, without standard “user management” authentication), make sure that `auth.xml` contains only the `pam` value:

   ```
   <list>
     <value>pam</value>
   </list>
   ```

7. Deploy the `midas-wapam-sa.zip` service assembly by running the following command:

   ```
   cp /opt/OV/OMU/adminUI/assemblies/midas-wapam-sa.zip /opt/OV/OMU/adminUI/deploy
   ```

8. Restart the WebApp by running the following command:

```
/opt/OV/OMU/adminUI/adminui restart
```

The following is a test example (on Linux):

```
# export
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/opt/OV/OMU/adminUI/lib/midas
# echo $LD_LIBRARY_PATH
# /opt/OV/OMU/adminUI/adminui restart
# /opt/OV/nonOV/jre/b/bin/java -cp
/opt/OV/OMU/adminUI/lib/cli/midas_cli.jar:/opt/OV/OMU/adminUI/work
/service-assemblies/midas-wapam/version_1/sus/servicemix-lwcontain
er/midas-pam/lib/jpam-1.1.jar:/opt/OV/OMU/adminUI/lib/commons-logg
ing-1.1.jar com/bes/itm/comp/usermgmt/TestPam opc_adm opc_pam
```

LDAP Authentication

To authenticate Administration UI users using LDAP, no extra software is needed because the Administration UI already includes the Acegi Security System for Spring Project open-source component. For details about this component, see the following URL:

http://acegisecurity.org

Note: Currently, only basic authentication of user accounts is supported. No additional LDAP features such as group membership can be used.

When configuring LDAP authentication, choose one of the following two methods:

- "Configuring LDAP Authentication Without Active Directory" on page 510
- "Configuring LDAP Authentication By Using Active Directory" on page 514

Note: To check the configuration values of your LDAP authentication configuration, you can use either Active Directory Users and Computers or the Apache Directory Studio open-source application.

Configuring LDAP Authentication Without Active Directory

To configure LDAP authentication without Active Directory, follow these steps:

1. Add all LDAP users that you want to authenticate to the Administration UI, and then set the corresponding user roles.
2. Configure the desired LDAP server in the `ldap.properties` file (`/opt/OV/OMU/adminUI/conf/ldap.properties`) by following these steps:
   a. Configure a URL pointing to the desired LDAP server.

      For example:

      ```
      # The LDAP URL
      # Format: ldap://<host>:<port>/<base dn>
      ```

```
# Format: ldaps://<host>:<port>/<base dn>
ldap.url=ldap://astrid:389/dc=hp,dc=com
#ldap.url=ldaps://astrid:636/dc=hp,dc=com
```

For both unencrypted and encrypted access, use `ldap.url=ldap://`.

Note: Make sure that you update the URL and the LDAP port based on your LDAP settings, as well as check your distinguished name (DN).

This example is used for the following scenario:

```
<host> : astrid
<port> : 389
<base dn>: dc=hp,dc=com
Full URL: ldap://astrid:389/dc=hp,dc=com
```

In this instance, `dc=hp,dc=com` is the DN of the LDAP node that is marked as the initial context for LDAP operations. All subsequent LDAP operations (for example, `ldapsearch`) are performed on the subtree of that node.

Caution: Because the LDAP configuration is environment specific, make sure that you consult your LDAP administrator during the configuration process.

b. Continue with entering the log-on credentials. For example:

```
# Manager DN for login
ldap.managerDn=cn=Administrator,dc=hp,dc=com
# Manager password
ldap.managerPassword=******
```

In this instance, the `ldap.ManagerDn` property is the DN of the entry that is used to perform the BIND (authenticate) operation required for other LDAP operations (for example, `Search`, for the Administration UI). Keep in mind that the value of `ldap.managerPassword` must correspond to the password assigned to this entry.

c. Make sure that the LDAP authentication mode is set to the default value (that is, `BIND_WITH_DN`).

Note: The LDAP authentication mode can also be set to `USER_SEARCH`, but it is highly recommended to use the default value.

With the default mode, usually no further configuration changes are necessary, so you can leave everything else commented out as shown in the following example:

```
# The mode which is used for the authentication
# Allowed values are:
# BIND_WITH_DN:Use the authenticationDnPatterns for
identifying a user
# USER_SEARCH : Use the authenticationSearchBase and
```

```
# authenticationSearchFilter for identifying a user
ldap.authenticationMode=BIND_WITH_DN
```

d. Add patterns for searching the users:

   `ldap.authenticationDnPatterns=sn={0},ou=People`

   In this instance, multiple patterns can be added, but they must be separated by vertical bars (|). These patterns represent Relative Distinguished Names (RDNs) that are relative to a root node configured in the `ldap.url` property. During authentication, `{0}` is replaced with a supplied user name.

   For example, if a user wants to log on with the `admin` user name, `ldapsearch` searches for an entry with the following DN (this search is based on the previously specified configuration settings):

   `sn=admin,ou=People,dc=hp,dc=com`

e. Verify the certificate. There are two possible scenarios:

   - The certificate originates from a proper third-party certification authority such as Verisign.

     In this case, no other change should be necessary.

   - A secure encrypted URL string is used, but without a certificate from a proper third-party certification authority.

     In this case, it is necessary to import the certificate into the local Administration UI truststore by following these steps:

     A. Configure the path to the truststore file and the truststore password as shown in the following example:

     ```
     # The path to the truststore for trusted certificates
     for secure LDAP
     ldap.truststore=conf/servicemix/truststore.jks
     # The truststore password for secure LDAP
     ldap.trustPassword=password
     ```

     B. Import the `.cer` format certificate by running the following command:

     ```
     <JRE_path>/bin/keytool -import -alias ldapserver_a -keystore
     /opt/OV/OMU/adminUI/conf/servicemix/truststore.jks -file /tmp/ldap_
     server.cer
     ```

     The *<JRE_path>* is `/opt/OV/nonOV/jre/b`.

     C. Answer the following questions:

     ```
     Enter keystore password: *******
     [...]
     Trust this certificate? [no]: yes
     ```

     The default password for the Administration UI truststore is `password`.

3. Activate the external authentication service in the `auth.properties` file by following these steps:
   a. Open the `auth.properties` file with the vi editor by running the following command:

      ```
      vi /opt/OV/OMU/adminUI/conf/auth.properties
      ```

   b. Edit the `auth.properties` file so that it contains the following:

      ```
      # configuration properties for authentication and
      authorization components
      #auth-filter.enabled=false
      usermodel-router.authResource=file:conf/auth.xml
      # eof
      ```

4. Switch the Administration UI to LDAP authentication by configuring the `auth.xml` file.

   The following is an example file:

   ```
   <?xml version="1.0" encoding="UTF-8"?>
   <!DOCTYPE beans PUBLIC "-//SPRING//DTD BEAN//EN" "http://
   www.springframework.org/dtd/spring-beans.dtd">
   <beans>
    <bean id="targetServices" class="java.util.ArrayList">
    <constructor-arg>
    <list>
    <value>ldap</value>
    <value>usermgmt</value>
    </list>
    </constructor-arg>
    </bean>
   </beans>
   ```

   The Administration UI tries to use the LDAP server for logon. If this authentication fails, the Administration UI tries standard “user management” authentication.

   If you want to set up only LDAP authentication (that is, without standard “user management” authentication), make sure that `auth.xml` contains only the `ldap` value:

   ```
   <list>
    <value>ldap</value>
   </list>
   ```

   Independently of whether LDAP or LDAPS is used, the default value must be `ldap`.

5. Deploy the `midas-waldap-sa.zip` service assembly by running the following command:

   ```
   cp /opt/OV/OMU/adminUI/assemblies/midas-waldap-sa.zip
   /opt/OV/OMU/adminUI/deploy
   ```

6. Restart the WebApp by running the following commands:

   ```
   /opt/OV/OMU/adminUI/adminui clean
   /opt/OV/OMU/adminUI/adminui start
   ```

Configuring LDAP Authentication By Using Active Directory

To configure LDAP Authentication using Active Directory, follow these steps:

1. Add all LDAP users that you want to authenticate to the Administration UI, and then set the corresponding user roles.
2. Configure the desired LDAP server in the `ldap.properties` file (`/opt/OV/OMU/adminUI/conf/ldap.properties`) by following these steps:
   a. Configure a URL pointing to the desired LDAP server.

      For example:

      ```
      # The LDAP URL
      # Format: ldap://<host>:<port>/<base dn>
      # Format: ldaps://<host>:<port>/<base dn>
      ldap.url=ldap://electron:389/DC=eledc08,DC=lan
      #ldap.url=ldaps://electron:389/DC=eledc08,DC=lan
      ```

      For both unencrypted and encrypted access, use `ldap.url=ldap://`.

      Note: Make sure that you update the URL and the Active Directory port based on your LDAP settings, as well as check your DN.

      This example is used for the following scenario:

      ```
      <host> : electron
      <port> : 389
      <base dn> : DC=eledc08,DC=lan
      Full URL : ldap://electron:389/DC=eledc08,DC=lan
      ```

      In this instance, `DC=eledc08,DC=lan` is the DN of the LDAP node that is marked as the initial context for LDAP operations. All subsequent LDAP operations (for example, `ldapsearch`) are performed on the subtree of that node.

      Caution: Because the LDAP configuration is environment specific, make sure that you consult your LDAP administrator during the configuration process.

   b. Continue with entering the log-on credentials. For example:

      ```
      # Manager DN for login
      ldap.managerDn=CN=Administrator,DC=eledc08,DC=lan
      # Manager password
      ldap.managerPassword=******
      ```

      In this instance, the `ldap.ManagerDn` property is the DN of the entry that is used to perform the BIND (authenticate) operation required for other LDAP operations (for example, `Search`, for the Administration UI). Make sure that the value of `ldap.managerPassword` corresponds to the password assigned to this entry.

c. Set the LDAP authentication mode to `USER_SEARCH` and, depending on the Active Directory server configuration, define the log-on name field as shown in the following example:

```
# The mode which is used for the authentication
# Allowed values are:
# BIND_WITH_DN : Use the authenticationDnPatterns for
identifying a user
# USER_SEARCH : Use the authenticationSearchBase and
# authenticationSearchFilter for identifying a user
ldap.authenticationMode=USER_SEARCH
# The search base for searching users for authentication
# This property is used in combination with the
# ldap.authenticationSearchFilter
# and is used e.g. for a Active Directory search
ldap.authenticationSearchBase=
# The filter for searching users for authentication
# This property is used in combination with the
ldap.authenticationSearchBase
# and is used e.g. for a Active Directory search
ldap.authenticationSearchFilter=(sAMAccountName={0})
ldap.authenticationSearchScope=SUBTREE_SCOPE
```

Note: `ldap.authenticationSearchBase` enables you to specify the location for user search. You can set the value for `ldap.authenticationSearchBase` as the name of the directory for user search, or leave the value empty. When no value is assigned, user search happens in the entire domain specified by `base_dn` from `ldap.url`.

Note that the value for `ldap.authenticationSearchBase` cannot be empty when the value for `ldap.authenticationSearchScope` is set as `ONELEVEL_SCOPE`.

Note: `ldap.authenticationSearchScope` enables you to restrict the scope of user search to a location. You can set the value for `ldap.authenticationSearchScope` as `ONELEVEL_SCOPE` or `SUBTREE_SCOPE`.

`ONELEVEL_SCOPE` enables user search in the directory specified by `ldap.authenticationSearchBase`.

`SUBTREE_SCOPE` enables user search in the:

- Directory specified by `ldap.authenticationSearchBase`
- Subdirectories of the directory specified by `ldap.authenticationSearchBase`

d. Verify the certificate. There are two possible scenarios:

- The certificate originates from a proper third-party certification authority such as Verisign.

  In this case, no other change should be necessary.
- A secure encrypted URL string is used, but without a certificate from a proper third-party

certification authority.

In this case, it is necessary to import the certificate into the local Administration UI truststore by following these steps:

A. Configure the path to the truststore file and the truststore password as shown in the following example:

```
# The path to the truststore for trusted certificates
for secure LDAP
ldap.truststore=conf/servicemix/truststore.jks
# The truststore password for secure LDAP
ldap.trustPassword=password
```

B. Import the `.cer` format certificate by running the following command:

```
<JRE_path>/bin/keytool -import -alias ldapserver_a -keystore
/opt/OV/OMU/adminUI/conf/servicemix/truststore.jks -file /tmp/ldap_
server.cer
```

The *<JRE_path>* is `/opt/OV/nonOV/jre/b`.

C. Answer the following questions:

```
Enter keystore password: *******
[...]
Trust this certificate? [no]: yes
```

The default password for the Administration UI truststore is `password`.

3. Activate the external authentication service in the `auth.properties` file by following these steps:
   a. Open the `auth.properties` file with the `vi` editor by running the following command:

      ```
      vi /opt/OV/OMU/adminUI/conf/auth.properties
      ```

   b. Edit the `auth.properties` file so that it contains the following:

      ```
      # configuration properties for authentication and
      authorization components
      #auth-filter.enabled=false
      usermodel-router.authResource=file:conf/auth.xml
      # eof
      ```

4. Switch the Administration UI to LDAP authentication by configuring the `auth.xml` file.

   The following is an example file:

   ```
   <?xml version="1.0" encoding="UTF-8"?>
   <!DOCTYPE beans PUBLIC "-//SPRING//DTD BEAN//EN" "http://
   www.springframework.org/dtd/spring-beans.dtd">
   <beans>
    <bean id="targetServices" class="java.util.ArrayList">
    <constructor-arg>
    <list>
    <value>ldap</value>
   ```

```
 <value>usermgmt</value>
 </list>
 </constructor-arg>
 </bean>
</beans>
```

The Administration UI tries to use the Active Directory server for logon. If this authentication fails, the Administration UI tries standard “user management” authentication.

If you want to set up only Active Directory authentication (that is, without standard “user management” authentication), make sure that `auth.xml` contains only the `ldap` value:

```
<list>
 <value>ldap</value>
</list>
```

Independently of whether LDAP or LDAPS is used, the default value must be `ldap`.

5. Deploy the `midas-waldap-sa.zip` service assembly by running the following command:

   ```
   cp /opt/OV/OMU/adminUI/assemblies/midas-waldap-sa.zip
   /opt/OV/OMU/adminUI/deploy
   ```

6. Restart the WebApp by running the following commands:

   ```
   /opt/OV/OMU/adminUI/adminui clean
   /opt/OV/OMU/adminUI/adminui start
   ```

Daylight Saving Time (DST) Patches

For JDK DST changes or JDK hotfixes, you must update the HP Operations management server. The JDK updates or hotfixes will not be included in any Administration UI patch.

The Administration UI uses OvJREB from the Accessories patch. You can get the updates by downloading Java Time Zone Updater tool 1.3.57 that is available at:

http://www.oracle.com/technetwork/java/javase/downloads/tzupdater-download-513681.html

Note: On Linux, extract `tzupdater.jar` from the `.ZIP` file to the `/` (`root`) directory.

To check your existing Java version, run the following command:

```
/opt/OV/nonOV/jre/b/bin/java -version
```

Your JRE image is bundled with the HP Operations management server after the installation. To update it, perform the following steps:

1. Stop the HP Operations management server. Run:

   ```
   /opt/OV/bin/ovc -stop
   ```

2. Invoke the Updater tool. Run the following command:

```
/opt/OV/nonOV/jre/b/bin/java -jar <tzupdater_PATH>/tzupdater.jar -u -v
```

3. Verify whether the update was successful. Run the following command:

   ```
   /opt/OV/nonOV/jre/b/bin/java -jar <tzupdater_PATH>/tzupdater.jar -t -v
   ```

4. Start the HP Operations management server. Run the following command:

   ```
   /opt/OV/bin/ovc -start
   ```

Part IV: Interoperability and Integration

HP Operations Manager (HPOM) supports interoperability in flexible management environments. You can distribute responsibility for managed nodes among management servers that have different operating systems. Flexible management can be agent-based and server-based.

HPOM enables integration of different applications, including third party applications. You can also configure HPOM to external trouble-ticket systems and notification services.

For detailed information on interoperability and integration, see:

- "HPOM Interoperability" on page 520
- "Application Integration with HPOM" on page 537
- "Notification Services and Trouble-Ticket Systems" on page 558

Chapter 11: HPOM Interoperability

In This Chapter

In this chapter, you can find the information about the following topics:

To find out about the main considerations and actions that must be followed for HPOM communication to work in a flexible management environment with NAT, see the *Firewall Concepts and Configuration Guide*.

For any additional information regarding interoperability between HPOM on UNIX or HPOM on Linux and HPOM for Windows, see HPOM on Windows online help that you can find at the following location:

https://softwaresupport.hp.com/group/softwaresupport/search-result?keyword=

Interoperability in Flexible Management Environments

In a flexible management environment, you can spread responsibility for managed nodes over multiple management servers, thereby enabling managed nodes to send messages to various management servers according to the time of a day, a location, or a subject of a message.

All participating HP Operations management servers should have the same major version of HPOM. However, there may be situations where one or more management servers still run on an older version (for example, when you upgrade your HPOM environment to a higher version and some management servers are not upgraded yet).

It is recommended that you upgrade all HP Operations management servers and managed nodes to the latest version of HPOM in a timely manner. Mixed version environments should remain a temporary solution.

Interoperability Between HPOM on UNIX or HPOM on Linux and HPOM on Windows

The HP Operations management server is available in three versions: a UNIX version (running on HP-UX and Solaris), a Linux version, and a Windows version. HPOM's flexible management functionality enables you to configure management servers or managed nodes to forward or send messages to different management servers.

HPOM provides the following possibilities for exchanging messages and configuration:

- Message forwarding:

  HPOM on UNIX or HPOM on Linux management servers can forward messages to HPOM on Windows management servers and vice versa.

  For more information, see "Server-based Flexible Management" on page 523.

- Message sending:

  HP Operations agents can send messages in the following directions:

- HPOM on UNIX or HPOM on Linux agents to HPOM on Windows management servers
- HPOM on Windows agents to HPOM on UNIX or HPOM on Linux management servers

  For more information, see "Agent-based Flexible Management" on page 522.

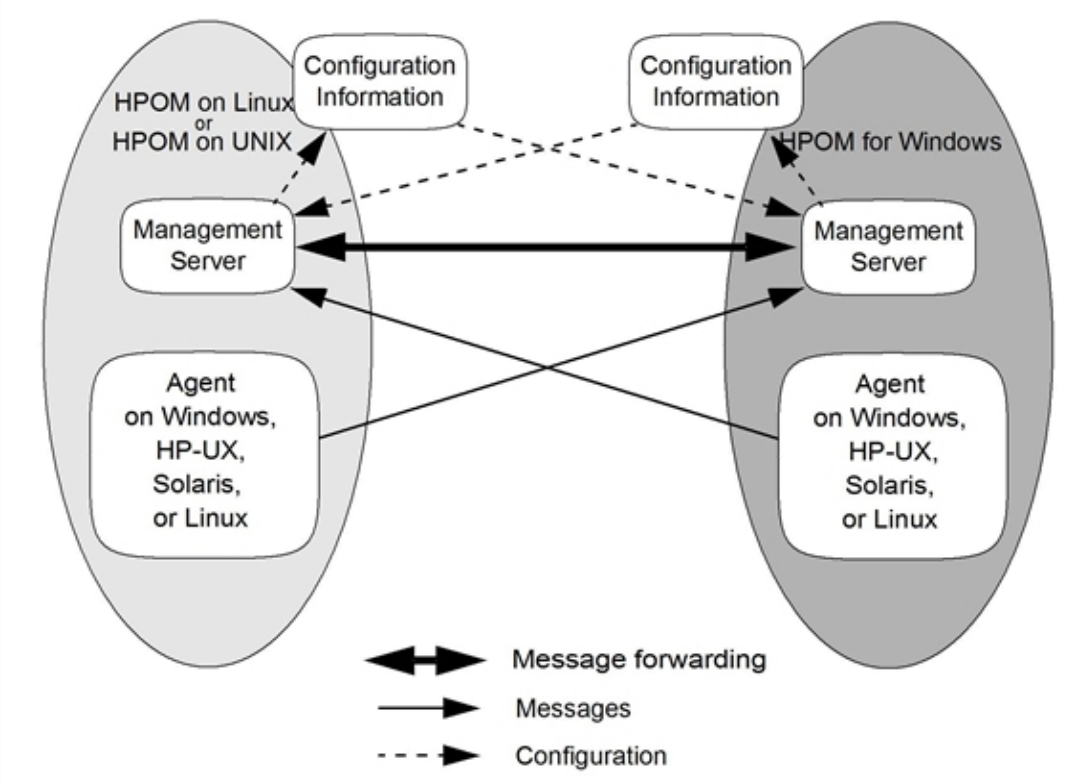
- Configuration:

  You can exchange HPOM configuration data such as policies and nodes between HPOM on UNIX or HPOM on Linux and HPOM on Windows by using different command line interfaces.

  For more information, see "Configuration Data Exchange" on page 525.

Figure 20 shows various communication paths between HPOM on UNIX or HPOM on Linux and HPOM on Windows.

Figure 20: HPOM on UNIX or HPOM on Linux and HPOM on Windows Interoperability



Agent-based Flexible Management

Agent-based flexible management enables you to configure managed nodes to send messages to different management servers based on time and message attributes. This functionality enables you to manage your worldwide network more effectively across time zones (for example, by using follow-the-sun control). It also enables you to increase efficiency (for example, by creating competence centers).

Note: Agent-based flexible management is not about forwarding messages from one management server to another, but about specifying which messages from a managed node should be sent to a specified management server.

By using agent-based flexible management, you can configure managed nodes to communicate directly with servers other than the primary management server. You can configure your managed nodes to communicate with the management servers of your choice anywhere in your network.

For detailed information about configuring message sending, see "Configuring Message Sending" on page 522.

For detailed information about agent-based flexible management and the follow-the-sun concept, see HPOM on Windows online help and the *HPOM Concepts Guide.*

Configuring Message Sending

To configure the HPOM on UNIX or HPOM on Linux managed node to send messages based on time and message attributes to the HPOM on Windows management server, follow these steps:

1. Synchronize trusted certificates on management servers (for example, on management server M1 and management server M2), where management server M1 gets a root certificate of management server M2 and management server M2 a root certificate of management server M1.

   To synchronize the trusted certificates on the management servers, follow these steps:

   a. On management server M1, run the following command:

      ```
      ovcert -exporttrusted -ovrg server -file <my_file>
      ```

   b. Copy `<my_file>` to management server M2 (for example, by using `ftp`).

   c. On management server M2, run the following command:

      ```
      ovcert –importtrusted –file <cert_file>
      ```

      ```
      ovcert –importtrusted –file <cert_file> -ovrg server
      ```

   d. Repeat the same procedure for management server M2.

   e. Verify that management server M1 and management server M2 have the root certificate of each other by running the following command on both management servers:

```
ovcert -list
```

Two trusted certificates should be listed.

2. Update the local root certificates on each managed node (including the local agent on each management server) by running the following command:

   ```
   ovcert -updatetrusted
   ```

3. Make sure that the management servers participating in the message sending operation recognize each other by adding them to the node bank of each participating management server.
4. Make sure that the OvCoreID of the node is set correctly in the node bank.

   To check the OvCoreID, run the following command:

   ```
   /opt/OV/bin/OpC/utils/opcnode -list_id node_name=<node_name>
   ```

   To set the OvCoreID in the database, run the following command:

   ```
   /opt/OV/bin/OpC/utils/opcnode -chg_id node_name=<node_name> id=<OvCoreID>
   ```

5. Access the appropriate example policy in the following directory:

   ```
   /etc/opt/OV/share/conf/OpC/mgmt_sv/tmpl_respmgrs
   ```

6. Copy the example policy to the working directory (for example, `/etc/opt/OV/share/conf/OpC/mgmt_sv/work_respmgrs`).

   Note: The name of the file should be either `allnodes` or *<hex_ip_addr_of_agent>*.

7. Modify the example policy to suit your environment, and then check the syntax by using the `opcmomchk(1)` tool.
8. Copy the modified file to the following directory so that it can be accessed by HPOM:

   ```
   /etc/opt/OV/share/conf/OpC/mgmt_sv/respmgrs
   ```

9. Start policy distribution to concerned managed nodes by using the `opcragt` command.

Note: HPOM automatically distributes the policy to the specified managed nodes as part of a normal distribution process.

For detailed information about how to configure the HPOM on Windows managed node to send messages based on time and message attributes to the HPOM on UNIX or HPOM on Linux management server, see HPOM on Windows online help.

Server-based Flexible Management

Server-based flexible management enables you to configure message forwarding between multiple HP Operations management servers (HPOM on UNIX or HPOM on Linux and HPOM on Windows), and also to forward message operations (such as acknowledge, own, and severity change) for forwarded messages. This means that messages are synchronized between the management servers

even when a message is changed on one of the management servers. Management servers can also forward action responses, which contain information about the success or failure of operator-initiated and automatic actions.

For detailed information about configuring message forwarding, see “"Configuring Message Forwarding" on page 524.”

Configuring Message Forwarding

To configure the HPOM on UNIX or HPOM on Linux management server to forward messages to the HPOM on Windows management server, follow these steps:

1. Synchronize trusted certificates on management servers (for example, on management server M1 and management server M2), where management server M1 gets a root certificate of management server M2 and management server M2 a root certificate of management server M1.

   To synchronize the trusted certificates on the management servers, follow these steps:

   a. On management server M1, run the following command:

      ```
      ovcert -exporttrusted -ovrg server -file <my_file>
      ```

   b. Copy *<my_file>* to management server M2 (for example, by using `ftp`).

   c. On management server M2, run the following command:

      ```
      ovcert –importtrusted –file <cert_file>
      ```

      ```
      ovcert –importtrusted –file <cert_file> -ovrg server
      ```

   d. Repeat the same procedure for management server M2.

   e. To verify that management server M1 and management server M2 have the root certificate of each other, run the following command on both management servers:

      ```
      ovcert -list
      ```

      Two trusted certificates should be listed.

2. Update the local root certificates on each managed node (including the local agent on each management server) by running the following command:

   ```
   ovcert -updatetrusted
   ```

3. Make sure that the management servers participating in the message forwarding operation recognize each other by adding them to the node bank of each participating management server.

4. Change to the directory that contains example policies for configuring server-based message forwarding. Run the following command:

   ```
   cd /etc/opt/OV/share/conf/OpC/mgmt_sv/tmpl_respmgrs
   ```

5. As the `root` user, copy the `msgforw` example policy to the working directory.

   For example, run the following command:

```
cp msgforw /etc/opt/OV/share/conf/OpC/mgmt_sv/work_respmgrs/msgforw
```

6. Modify the example policy to suit your environment, and then check the syntax by using the `opcmomchk(1)` tool.
7. As the `root` user, copy the validated file to the following configuration directory:

   `/etc/opt/OV/share/conf/OpC/mgmt_sv/respmgrs`
8. After modifying the example policy, run the `ovconfchg` command without any options to reload the new configuration:

   `/opt/OV/bin/ovconfchg`

For more information, see the *opcmom* manual page, which describes the HPOM on UNIX's flexible management functionality.

For detailed information about how to configure the HPOM on Windows management server to forward messages to the HPOM on UNIX or HPOM on Linux management server, see HPOM on Windows online help.

Configuration Data Exchange

In an environment with multiple management servers, you must configure each managed node on every management server that may receive the managed node's messages. Management servers discard messages if they originate from unknown managed nodes. This applies for both agent-based and server-based flexible management.

HPOM's configuration data exchange functionality provides flexible options for data exchange. You should plan how to create, maintain, and distribute configuration data between multiple management servers.

The general rule when using command line interfaces to exchange configuration data is that you first download the data to files on one management server and then copy the files to another management server and upload the data. When exchanging configuration data, you must use different command line interfaces for working with HPOM on UNIX or HPOM on Linux and HPOM on Windows management servers.

For downloading policies or nodes from the HPOM on UNIX or HPOM on Linux management server, the `opccfgdwn` command line interface is used. You can download configuration data in one of the following ways:

- *By using the Administration UI (recommended):*

  In the Administration UI, select the configuration data you want to download.
- *Directly:*

  Create a download specification file, and then run `opccfgdwn`.

For more information, see "Examples of the Configuration Data Exchange" on page 526.

Table 38 summarizes which command line interfaces you must use. It also shows where it is recommended to use the Administration UI.

Table 38: Interfaces for Configuration Data Exchange

Configuration Data	Download Upload	UNIX or Linux to Windows	Windows to UNIX or Linux
Nodes	Download Upload	Administration UI *(recommended)* / `opccfgdwn` `ovowconfigexchange`	`ovowconfigexchange` `opccfgupld`
Policies	Download Upload	Administration UI *(recommended)* / `opccfgdwn` `ovpmutil`	`ovpmutil` `opcpolicy`
Instruction Text	Download Upload	Only possible by exchanging policies	`ovowconfigexchange` `opcpolicy`
Services	Download Upload	Not supported	`ovpmutil` `opcservice`
Instrumentation	Download Upload	Copy from file system	Copy from file system

Examples of the Configuration Data Exchange

This section uses examples to show how to exchange the following types of configuration data:

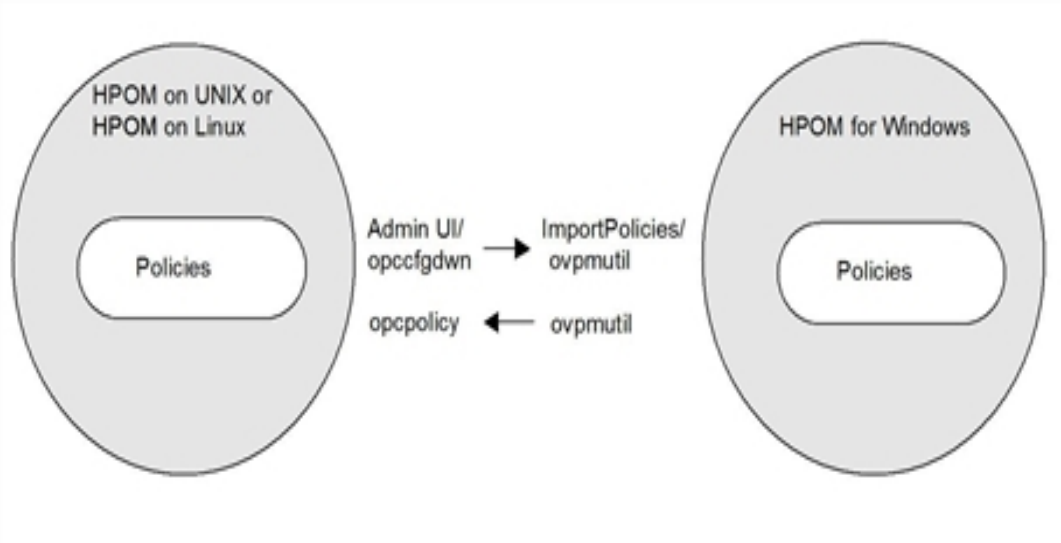
- Policies
- Nodes
- Services

Because HP supports integration between specific versions of HP Operations management servers only, there are some restrictions on the data that you can exchange between HPOM on UNIX or HPOM on Linux and HPOM on Windows. For up-to-date details of supported integrations, see the support matrix at the following location:

http://support.openview.hp.com/selfsolve/document/KM323488

Policy Exchange

This example describes how to exchange policy configuration data between HPOM on UNIX or HPOM on Linux and HPOM on Windows management servers.



When exchanging policy configuration data between the HPOM on UNIX or HPOM on Linux and HPOM on Windows management servers, do as follows:

- **Download policies from the HPOM on UNIX or HPOM on Linux management server and upload them to the HPOM on Windows management server:**
  a. To download policies from the HPOM on UNIX or HPOM on Linux management server, choose one of the following options:
     - Administration UI (the recommended option)

       In the Administration UI, select the policies and/or policy groups that you want to download, and then select **Download...** from the drop-down menu.
     - `opccfgdwn` command line interface

       For detailed information about using the `opccfgdwn` command line interface to download policies, see the *opccfgdwn(1m)* manual page.
  b. To upload policies to the HPOM on Windows management server, use either the `ImportPolicies` or `ovpmutil` command (depending on the version of HPOM).

For detailed information, see HPOM on Windows online help.

- **Download policies from the HPOM on Windows management server and upload them to the HPOM on UNIX or HPOM on Linux management server:**
  a. To download policies from the HPOM on Windows management server, run the following command:

     ```
     ovpmutil CFG POL DNL <targetdir>/p <policy_group_path> /8x /instrum
     ```

For example:

```
ovpmutil CFG POL DNL adv_hpux /p "SPI for Unix OS\en\HP-UX\Advanced HP-UX
Policies" /8x /instrum
```

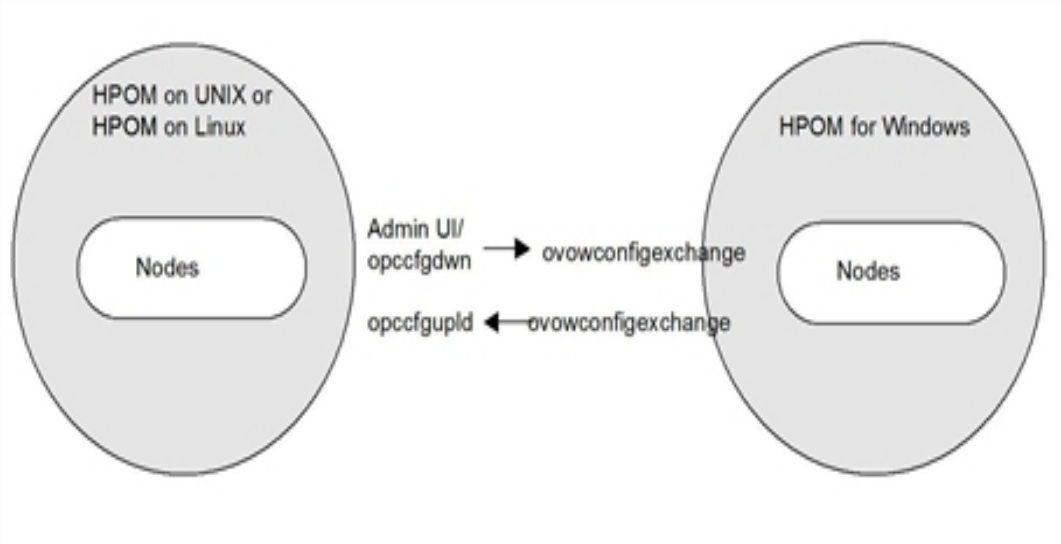
b. To upload policies to the HPOM on UNIX or HPOM on Linux management server, run the following command:

```
opcpolicy -upload mode=replace dir=<directory>
```

Note: When uploading policies, instrumentation and categories are also uploaded.

Node Exchange

This example describes how to exchange node configuration data between HPOM on UNIX or HPOM on Linux and HPOM on Windows management servers.



When exchanging node configuration data between the HPOM on UNIX or HPOM on Linux and HPOM on Windows management servers, do as follows:

- **Download nodes from the HPOM on UNIX or HPOM on Linux management server and upload them to the HPOM on Windows management server:**
  a. To download nodes from the HPOM on UNIX or HPOM on Linux management server, choose one of the following options:
     - Administration UI (the recommended option)

       In the Administration UI, select the nodes and/or node groups that you want to download, and then select **Download...** from the drop-down menu.
     - `opccfgdwn` command line interface

For detailed information about using the `opccfgdwn` command line interface to download nodes, see the *opccfgdwn(1m)* manual page.

b. To upload nodes to the HPOM on Windows management server, use the `ovowconfigexchange` command.

For detailed information, see HPOM on Windows online help.

- **Download nodes from the HPOM on Windows management server and upload them to the HPOM on UNIX or HPOM on Linux management server:**

a. To download nodes from the HPOM on Windows management server, run the following command:

```
ovowconfigexchange -ent NODES -dnl nodes -dest_codeset utf8
```

b. To upload nodes to the HPOM on UNIX or HPOM on Linux management server, run the following command:

```
opccfgupld -add <path>/nodes
```

Service Exchange

Note that downloading services from the HPOM on UNIX or HPOM on Linux management server and uploading them to the HPOM on Windows management server is not supported.

To download services from the HPOM on Windows management server and upload them to the HPOM on UNIX or HPOM on Linux management server, follow these steps:

1. To download services from the HPOM on Windows management server, run the following command:

```
ovpmutil CFG XML DNL <configfile> /p <serviceID>
```

2. To upload services to the HPOM on UNIX or HPOM on Linux management server, run the following command:

```
opcservice –add <configfile>
```

Online Configuration Synchronization

In a multiple management server environment, you can synchronize HP Operations management server data between a primary management server and one or more backup management servers by using the `opccfgsync` command line tool.

During synchronization of the configuration data, the following major processes take place:

- Configuration data download on the primary management server
- Configuration data transfer to one or more backup management servers
- Configuration data upload on one or more backup management servers

Caution: Before you start with synchronization of the configuration data, you must register all involved management servers.

For detailed information, see "Management Server Registration" on page 532.

Configuration synchronization is fully customized by using scenario files. A scenario file contains the rules that specify how to download and upload configuration. To download, transfer, and upload the configuration data for all registered management servers, run the following command:

```
/opt/OV/bin/OpC/utils/opccfgsync –sync <scenario>
```

For example:

```
/opt/OV/bin/OpC/utils/opccfgsync -sync OnlineBackup
```

After you run this command, the configuration download on the local management server is performed. The scenario file is read line by line and the actions are performed according to the configuration blocks and their keywords. During the configuration download, the scenario for the configuration upload is also generated. For detailed information about scenario files, see "Scenario Files" on page 533.

The default location for downloading and uploading the configuration data is as follows:

```
/var/opt/OV/share/tmp/OpC/cfgsync/
```

Each configuration download is stored in the `BASE_DIR` subdirectory that looks as follows:

```
<scenario_name>_<current_date>_<current_time>
```

This download path is specified inside the scenario file as `BASE_PATH`. After the configuration download is finished, an archive of `BASE_DIR` is made. When synchronizing the download configuration data to the backup management server, the data is transferred to `BASE_PATH` of the backup management server.

For each registered management server (except for the primary management server), `opccfgsync` performs the following two steps:

- Step 1: Transfers an archive to a selected management server.
- Step 2: Extracts an archive and performs the configuration upload as specified in the upload scenario file inside the archive.

For synchronizing HP Operations management server data between management servers, use the `opccfgsync` tool that can be found at the following location:

```
/opt/OV/bin/OpC/utils/
```

The syntax of the `opccfgsync` tool is as follows:

```
opccfgsync -sync <scenario>
           -download <scenario>
           -upload <upload_dir>
           -scenarios
           -servers
           -register [<server> [<comm_type> [<location>]]]
           -unregister [<server>
           -primary [<server>]
           -v
           -h|-\?|-help
```

You can use the following options with the `opccfgsync` tool:

`-sync <scenario>`	Downloads, transfers, and uploads configuration data for all registered management servers.
`-download <scenario>`	Downloads the configuration data on the local management server.
`-upload <upload_dir>`	Uploads the configuration directory on the local management server.
`-scenarios`	Lists all scenarios (the include files are excluded).
`-servers`	Lists registered management servers.
`-register`	Registers the management server[a], the communication type[b], and the location[c].
`-unregister`	Unregisters the management servera.
`-primary`	Sets the primary management servera.
`-v`	Shows the verbose output.
`-h|-\?|-help`	Shows the usage.

For detailed information about all `opccfgsync` options, see the *opccfgsync* manual page.

Communication Types

In a multiple management server environment, different communication types can be used between the selected management servers. When you perform the configuration upload on one or more remote management servers, you can use the following communication types:

[a]If the management server is not specified, the local host is used.

[b]If the communication type is not specified, the default is used—SSH.

[c]If the location is not specified, the default is used—the directory where configuration is downloaded and uploaded (that is, `/var/opt/OV/share/tmp/OpC/cfgsync`).

- OVDEPLOY

  To use this communication type, you must establish a trust relationship between management servers. Also make sure that L-Core processes run on all management servers. Otherwise, the configuration upload does not work.

  To check communication, run the following command:

  ```
  ovdeploy -cmd hostname -host <remote_server>
  ```

- REMSH

  To enable remote shell communication, you must add the nodes to the to the `.rhosts` system configuration file.

  To check communication, run the following command:

  ```
  rsh <remote_server> hostname
  ```

- SSH (the default communication type)

  To enable passwordless SSH communication, public keys between the nodes must be exchanged.

  To check communication, run the following command:

  ```
  ssh <remote_server> hostname
  ```

Management Server Registration

Configuration synchronization is performed from the primary management server to all other registered management servers. Therefore, before synchronizing HP Operations management server data between management servers, you must register all involved management servers. To do this, run the following command:

```
/opt/OV/bin/OpC/utils/opccfgsync -register [<server> [<comm_type> [<location>]]]
```

For example:

```
/opt/OV/bin/OpC/utils/opccfgsync -register serverA REMSH /tmp/CfgSync
```

When all the management servers are registered, set up the primary management server by using the following command:

```
/opt/OV/bin/OpC/utils/opccfgsync –primary <server_name>
```

For example:

```
/opt/OV/bin/OpC/utils/opccfgsync –primary serverA
```

The location where the data is stored is the same as the default location for the configuration download and upload, and is as follows:

```
/var/opt/OV/share/tmp/OpC/cfgsync
```

However, during the management server registration, you can choose a different location for downloading and uploading the configuration data.

Note: It is not possible to upload configuration on the primary management server.

If you want to disable configuration synchronization to the selected backup management server, unregister it by running the following command:

```
/opt/OV/bin/OpC/utils/opccfgsync -unregister <server_name>
```

For example:

```
/opt/OV/bin/OpC/utils/opccfgsync -unregister serverA
```

Scenario Files

A scenario file is a file in which the rules for downloading and uploading configuration are specified. Each scenario file is composed of configuration blocks. A configuration block starts with a line containing a keyword (specifying a block type) and a block name. The block name is used to describe the configuration block, so that each configuration block is uniquely identified. The configuration block usually ends with the `END` line. However, there are some configuration blocks that do not contain the `END` line. For details, see "Scenario File Syntax" on page 533.

The configuration block that contains the `END` line is structured as follows:

```
<keyword>:<block_name>
<block_data>
END
```

Caution: In some cases block names are used for file names. Therefore, make sure that block names do not contain spaces, tabs, or any other special characters.

You can find all scenario files at the following location:

```
/etc/opt/OV/share/conf/OpC/mgmt_sv/cfgsync/
```

Scenario File Syntax

Table 39 lists configuration blocks that contain the `END` line. It also describes the block data.

Table 39: Configuration Blocks with the `END` Line

Keyword	Block Data
`CONFIG:<name>`	Represents the contents of the configuration file used for `opccfgdwn`. The target directory for `opccfgdwn` is set to `$BASE_PATH/config/<name>`.

Configuration Blocks with the END Line, continued

Keyword	Block Data
FILES:*<name>*	Contains a list of files and directories that are archived and synchronized with other management servers. The selected files are stored in $BASE_PATH/files/*<name>*.tar.
EXEC:*<name>*	Contains shell script commands that are run during the download. The $BASE_PATH variable can be used inside this block (it contains the location of the configuration download). The contents of the EXEC block are written into $BASE_PATH/run/*<name>*.sh and run during the configuration download.
UPLOAD_EXEC:*<name>*	Contains shell script commands that are run during the upload on the remote management servers. The $BASE_PATH variable can be used inside this block (it contains the location of the configuration download). The contents of the UPLOAD_EXEC block are written into $BASE_PATH/run/*<name>*.sh and run during the configuration upload.
SQL:*<name>*	Contains SQL commands that are run during the configuration download. The opcdbpwd tool is used to run the SQL script. The output that is written into $BASE_PATH/sql/*<name>*.output can be analyzed by the EXEC or UPLOAD_EXEC block during the configuration upload.
UPLOAD_SQL:*<name>*	Contains SQL commands that are run during the configuration upload on the remote management servers. The contents of the UPLOAD_SQL block are written into $BASE_PATH/sql/*<name>*.sql and run by the opcdbpwd tool during the configuration upload.

Table 40 lists configuration blocks that do not contain the END line. It also describes the corresponding block names.

Table 40: Configuration Blocks Without the **END** Line

Keyword	Block Name Description
UPLOAD:*<upload_line>*	Specifies the commands for the configuration upload and is written into the upload configuration file.
INCLUDE:*<include_filename>*	This line is replaced with the contents of the specified include file.

Configuration Blocks Without the END Line, continued

Keyword	Block Name Description
CLEANUP:<*number_of_config_downloads/uploads*>	Specifies how many configuration downloads or uploads made by a specified scenario should be kept on the system. For example, if the number is three, last three configuration downloads or uploads made by this scenario stay on the system, while all others are removed.

Scenarios

The following default scenarios are available with the HP Operations management server:

- FullBackup

  With this scenario, a complete backup of the primary management server is performed. The configuration on backup management servers should be the same as the configuration on the primary management server. This scenario stops the backup management server, reinitializes the database, uploads the configuration, and then starts the backup management server.

  The following is backed up with this scenario:

- full configuration
- responsible manager configuration
- reports
- message history
- active messages
- configuration files from the following location:

  /etc/opt/OV/share/conf/OpC/mgmt_sv/

- OnlineBackup

  With this scenario, synchronization of most frequently modified data is performed. This scenario does not stop the backup management server.

  The following is backed up with this scenario:

- selected configuration
- message history
- active messages
- ConfigSyncData

This is the internal scenario that is used by the `opccfgsync` tool to check communication between management servers and to synchronize internal data such as configuration files and scenarios.

- `CertificateBackup`

  In this scenario, a certificate backup is performed on the primary management server and then the backup file is transferred to all backup management servers.

  The format of the backup file name is as follows:

  `CertBackup_<hostname>_<date>_<time>.tar`

Because all the scenarios are supported by the HP Operations management server, they can be customized during upgrades. To customize an existing scenario, copy it into a new scenario file. You can also create a new scenario by simply creating a new scenario file.

The tasks that are performed in several scenarios can be added to the include files. The `INCLUDE` keyword is used for inserting the contents of a specified include file into a scenario file. Include files are stored at the same location as scenario files and their names should begin with `include`.

Note: It is recommended to add `CLEANUP` blocks at the end of each scenario, so that a limited number of downloads remain on the system (occupying less space). To do this, use the following lines:

```
CLEANUP: <number_of_downloads>
UPLOAD:CLEANUP: <number_of_uploads>
```

Chapter 12: Application Integration with HPOM

In this Chapter

This chapter explains how to integrate applications into HP Operations Manager (HPOM). It also describes several integrations that are available with HPOM. For more information on a particular product which is integrated with the HPOM, see the documentation provided with this product. In this chapter, you can find detailed information about the following topics:

For more detailed information on the elements and the windows you can use to carry out the integration, see the *HPOM Concepts Guide*.

Application Integration

HP Operations Manager enables operators to invoke applications. Applications can include default tools installed with the product, any custom applications you write and integrate, or applications installed by a Smart Plug-in (SPI).

Application Assignment

You can assign a different set of applications to each operator, as needed.

Default HP Applications

If you have purchased an application that is already prepared for HPOM integration (for example, HP Data Protector), you can integrate it quickly and easily using the upload-configuration utility,

opccfgupld(1m). For more information about the options available with the opccfgupld command, run opccfgupld with the -h(elp) option.

Third-Party Applications

You can run third-party applications either before or after the configuration download or upload by using the opccfgdwn and opccfgupld command line tools . Running the 3rd party applications consists of the following steps:

1. opccfgdwn checks the following directories if they exist:
   - /etc/opt/OV/share/conf/OpC/mgmt_sv/integration/cfgdwn/pre/
   - /etc/opt/OV/share/conf/OpC/mgmt_sv/integration/cfgdwn/post/
2. opccfgupld checks the following directories if they exist:
   - /etc/opt/OV/share/conf/OpC/mgmt_sv/integration/cfgupd/pre/
   - /etc/opt/OV/share/conf/OpC/mgmt_sv/integration/cfgupd/post/
3. If any of the lowest-level directories contains executables, 3rd party applications are executed before (the pre directory) or after (the post directory) the configuration is downloaded or uploaded.

Note: The opccfgdwn and opccfgupld command line tools wait for each integration to finish or for a configured timeout to occur, which can be set by using the OPC_CFG_INTEG_TIMEOUT server configuration variable. For detailed information about this server configuration variable, see the *HPOM Server Configuration Variables* document.

The arguments that are passed to third-party applications by opccfgdwn and opccfgupld are as follows:

- Language in which the upload or the download is performed (the UTF-8 character set), for example: en_US.UTF-8
- Copies of the arguments that were used to call opccfgdwn or opccfgupld, for example: opccfgdwn -silent /tmp/user.dsf /tmp/opers

In this case, for example, the third-party application is run as follows:

<3rd_party_application_name> en_US.UTF-8 -silent /tmp/user.dsf /tmp/opers

Application Integration with HPOM Components

You can integrate applications into the following HPOM components:

- Java GUI
- Broadcasts
- Actions (automatic, operator-initiated, and scheduled)
- Monitoring
- Log-file encapsulation
- SNMP trap and message interception

Integrated Applications in the Java GUI

You can add your own applications, and assign them to an operator. The applications are then invoked when the operator clicks the application name under the `Tools` folder of the Java GUI `Object Pane`.

HPOM Application Integration

Typically, HPOM applications are utilities that provide services of a general nature, they help build a set of management tools. You can pass information (for example, selected nodes) as arguments to the applications. Users then start the applications by selecting them in the `Tools` folder of the Java GUI `Object Pane`.

Applications and application groups integrated into HPOM can be managed using the `opcappl` command line tool. For more information about command options and parameters, see the *opcappl(1m)* manual page. HPOM provides a selection of default applications and application groups.

Integrated Applications as Broadcast Commands

You can launch applications on multiple systems at the same time using the HPOM broadcast command facility in the Java GUI.

Integration Requirements

To launch an application on multiple systems, you must first meet the following requirements:

- UNIX systems:

  The application must be accessible from your *$PATH* settings.
- All systems:

  The path must be fully qualified on the `Broadcast Command` window.

Note: In all cases, the application you want to launch must be available on the managed node.

Application Distribution to Managed Nodes

You can distribute simple and widely used applications to managed nodes through HPOM. For details, see "HPOM Agent-Configuration Distribution" on page 211.

Integrated Applications as Actions

You may configure an application or script to run as an automatic action, operator-initiated action, or scheduled action:

- Automatic action:

  Action triggered by a message received in HPOM.
- Operator-initiated action:

  Action enabled by a message received in HPOM and executed by an operator.
- Scheduled action:

  Actions configured by the HPOM administrator. These actions execute a routine task at a preconfigured time.

Action Agent

Actions are always performed by the HPOM action agent, which operates as root on UNIX systems, as HP ITO Account on Windows systems. To be executed, the action must be available on the managed node.

Note: The HP ITO Account is part of the Administrator, Domain Administrator, and User Administrator groups. If an action is prohibited for one of these groups, the HP ITO Account is not able to perform that action.

Requirements for Integrating Applications as Actions

To integrate applications as action, the applications must meet the following requirements:

- UNIX systems:

  The application must be accessible from the *$PATH* settings of the root.
- All systems:

  The path must be fully qualified in the corresponding message condition.

Distributing Actions to Managed Nodes

You can distribute simple and widely used actions to managed nodes through HPOM. For details, see "HPOM Agent-Configuration Distribution" on page 211.

Integrating Monitor Applications

You can use applications for monitoring purposes by configuring them to deliver the monitored object status using the `opcmon(1)` command or `opcmon(3)` API.

Requirements for Integrating Monitored Applications

To integrate a monitored application into HPOM, the application must meet the following requirements:

- UNIX systems:

  The application must be accessible from the *$PATH* settings of the root.
- All systems:

  The path must be fully qualified in the corresponding message condition.

Note: In all cases, the application you want to launch must be available on the managed node.

Distributing Monitored Applications to Managed Nodes

You can distribute simple and widely used monitoring applications to managed nodes through HPOM. For details, see "HPOM Agent-Configuration Distribution" on page 211.

Monitoring Application Log Files

You can monitor applications by observing their log files. You can suppress log-file entries or forward them to HPOM as messages. You can also restructure these messages or configure them with HPOM-specific attributes.

Note: Most applications running on Windows systems use **Eventlogs**. The information in these databases can be extracted by the log-file encapsulator, but there are some differences in the setup procedure. For more information, see the *HPOM Concepts Guide*.

Intercepting Application Messages

To monitor applications, HPOM uses the following messages:

- Log files
- SNMP traps
- `opcmsg(1)` command
- `opcmsg(3)` API

Depending on how you have configured HPOM, you can suppress messages or forward them to HPOM. You can also restructure these messages or configure them with HPOM-specific attributes.

Message-Stream Interface API

You can use the Message-Stream Interface (MSI) API to register applications to receive messages on the management server. The MSI lets you plug in event-correlation engines and statistical-analysis tools to establish a link to other network and system-management applications.

Messages are intercepted before they are added to the HPOM database and before they are displayed in the HPOM message browsers.

Applications and Broadcasts on Managed Nodes

Before it starts an application or broadcast command on the managed node, HPOM verifies the profile of the executing user.

Restrictions on Applications and Broadcasts

The following restrictions apply to applications and broadcasts:

- Commands and applications:

  The HPOM action agent broadcasts commands and starts applications.

  Applications can be configured as follows:

  - `Window (Output Only)`

    A Window (Output Only) application is started by the agent on the managed node defined in the application. The output is channeled back to the Java GUI using HTTPS communication and displayed directly in the Java GUI when the command is finished.

    When using the Window (Output Only) application, keep in mind the following:

    - No password is needed because the application is started by the agent (unless the customized startup is used with a different user).
    - Starting the application will work through a firewall and proxies (HTTPS communication is used to transport the output back to the management server).

- The output is sent when the application is completely finished and before that you cannot see progress messages nor any input requests.

- `Window (Input/Output)`

  For an Window (Input/Output) application, the management server starts a terminal window to perform a remote logon to the managed node by using `rlogin`. The terminal emulator specified in the Virtual Terminal field of the node properties is used (that is, `dtterm`, `hpterm`, or `xterm`). In the terminal window, the `opcrlogin` program, which establishes an `rlogin` connection to the managed node and automatically specifies the user and the password if specified in the application, is started.

  Caution: On Linux, only `xterm` is available as a terminal emulator. If the specified terminal emulator is not available on the management server (for example, the default terminal emulator for HP-UX managed nodes is `dtterm`), it defaults to `xterm` on Linux.

  When using the Window (Input/Output) application, keep in mind the following:

  - An X redirection is required. This is specified through the Display parameter in the Java GUI log-on window. If your display station is Linux or UNIX, you can use it as your X-Windows display. If you use the Java GUI on a Windows PC, you can use a product such as Reflection X or Exceed to redirect the X output to the Windows system.
  - `rlogin` access from the management server to the managed node (including the management server itself if you start the Input/Output application on the management server) is required. On many systems, the remote logon to the managed node by using `rlogin` is usually disabled for security reasons.
  - Because the `rlogin` port is usually blocked by firewalls, the Window (Input/Output) application does not work for the managed nodes that are behind a firewall.
  - Proxy settings are not used for the remote logon to the managed node by using `rlogin`.
  - If the password is not provided when defining the application, the user must type the password to perform the remote logon to the managed node by using `rlogin`.

- `No Window` (for example, an X application)

  A No Window application is started by the agent on the managed node. The `DISPLAY` variable is set as specified in the Java GUI log-on window. This kind of application can be used for X applications such as Motif SAM on HP-UX, or simply a terminal emulator such as `xterm` that was started on the managed node directly. In this case, you must have an X redirection set up.

  When using the No Window application, keep in mind the following:

  - No password is needed because the application is started by the agent (unless the customized startup is used with a different user).

- Starting the application itself will work through a firewall. However, because the X protocol is usually blocked by firewalls, the X redirection will not work.
- Proxies are used for the execution, but not for the X redirection.

During the execution of a user profile, `stdin`, `stdout` and `stderr` are not available. For this reason, avoid commands reading from standard input or writing to standard output or error. In particular, avoid commands such as the following:

- `stty`
- `tset`
- Startup of window (input/output) applications

- Delays and inactivity:

  If a delay of more than two seconds occurs during application output or input, HPOM assumes that an error has occurred and stops application execution. For example, an HPOM error can occur if a program runs for more than two seconds without generating output.

Note: Applications do not require a separate terminal window.

User Profile Configuration

When setting up user profiles, take note of the following guidelines:

- User Input:

  Do not ask for specific user input in the profile. Instead, provide a default value that users can confirm by pressing **Return**. For example, the following examples show good and bad ways to write scripts that require user confirmation:

  - Not recommended:

    The following script for HP-UX 11.x produces an endless loop if no valid answer is specified:

```
#!/usr/bin/sh
TERM=""
while [ -z "${TERM}" ]
do
 echo "Type of terminal (hp|vt100): \c"
 read TERM
 if [ "${TERM}" != "hp" -a "${TERM}" != "vt100" ]
 then
   TERM=""
 fi
done
```

  - Recommended:

The following script shows the correct way to prompt the user to confirm a default value. If no valid answer is specified, a default value is used:

```
#!/usr/bin/sh
echo "Type of terminal (hp=default|vt100): \c"
read TERM
if [ "${TERM}" != "hp" -a "${TERM}" != "vt100" ]
then
  TERM=hp
fi
```

- Questions:

  Do not ask more than four questions in the user's profile. HPOM only answers up to four prompts with **Return**.

- Logout messages:

  Do not add a logout message to the user's profile. HPOM adds the message at the end of the application's output. In addition, do not use sequences of escape characters in the profile. Escape characters are also added to the application output, thereby garbling the output.

NNMi and HPOM

This section describes how to install, configure, and use the HP Network Node Manager i (NNMi) integration on HP Operations management servers. The HP NNMi integration forwards incidents from NNMi to the HPOM message browser and provides easy access to the NNMi console from within HPOM.

The NNMi integration software is installed automatically with HPOM. However, you need to configure the HPOM agent or web services implementation before you can use the integration, as follows:

- HP Operations agent implementation:

  The HPOM agent implementation of the NNMi–HPOM integration is available from NNMi version 8.12. This implementation is the preferred solution for integrating HPOM with NNMi. For more information about the agent implementation, see "NNMi Integration: Agent Implementation" on page 546.

- HPOM web services implementation:

  The *web service* implementation of the NNMi–HPOM integration is available from NNMi version 8.03. Note that the HPOM agent implementation is the preferred solution for integrating HPOM with NNMi. For more information about the web services implementation, see "NNMi Integration: Web Services Implementation" on page 549.

Note: If the HPOM-agent and the web services implementations of the NNMi–HPOM integration both forward messages to the same HPOM management server, you might not see all messages

from both implementations in the HPOM message browser. For this reason, HP does not support running both implementations of the NNMi–HPOM integration to the same HPOM management server concurrently.

You can see the forwarded NNMi incidents in the HPOM message browser. Because forwarded messages in the HPOM browser are associated with the original incidents reported in NNMi, you can launch the NNMi incident browser within HPOM and display the original incident.

Each NNMi incident has a unique identity, so that even where HPOM is consolidating events across multiple NNMi management servers, you can trace a particular incident back to its origin in NNMi and investigate it.

Some of the tools in the NNMi tools group are integrated by default into HPOM. This means you can access NNMi tools from nodes in the HPOM console, and from the active and history message browsers. For more information, see "NNMi Tools" on page 551.

Supported Versions

For up-to-date information about supported product versions for the NNMi–HPOM integration, see the support matrices at the following location:

http://support.openview.hp.com/selfsolve/document/KM323488

NNMi and HPOM must be installed on separate computer systems. The operating system of the NNMi management server and the HPOM management server are independent of each other. They can use the same operating system, although this not a requirement. For example, an NNMi management server can run on the HP-UX platform, while the HPOM management server runs on a Windows operating system.

NNMi Integration: Agent Implementation

When using the HPOM agent implementation, the HP NNMi integration forwards NNMi incidents as SNMPv2 traps to an HPOM agent on the NNMi management server. The HPOM agent filters the SNMPv2 traps and forwards them to the HPOM active messages browser. The configuration of the HPOM agent determines which HPOM management server receives the message.

The agent implementation of the HP NNMi–HPOM integration is available as of NNMi version 8.12. This implementation is the preferred solution for integrating HPOM with NNMi.

In NNMi, incidents can be created directly by NNMi (called management events) or created from SNMP traps. The NNMi northbound interface makes these incidents available as SNMPv2 traps. The HPOM agent listens at the northbound interface for these SNMPv2 traps. An SNMP Trap policy determines how the agent filters and processes the SNMPv2 traps.

The SNMP Trap policy is based on an SNMP Trap policy file that you create on the NNMi management server. The SNMP Trap policy file includes a policy condition for each of the management events and SNMP traps in the current NNMi configuration. The HPOM agent sends traps that pass the filters of the policy as messages to the HPOM management server.

The agent implementation of the HP NNMi–HPOM integration provides a one-way flow of NNMi incidents to HPOM. When the life cycle state of an incident changes to closed in NNMi, NNMi forwards a close event to HPOM. HPOM acknowledges the message for the original incident in the HPOM message browser. NNMi sends only one copy of each management event or SNMP trap to the HP Operations agent.

If you configure the HP NNMi–HPOM integration to forward all received SNMP traps and the HPOM management server receives SNMP traps directly from devices that NNMi manages, HPOM receives duplicate device traps. You can set the policies to correlate SNMP traps from NNMi with those that HPOM receives directly from managed devices.

You can see the forwarded NNMi incidents in the HPOM message browser. The tools in the NNMi tools group provide access to NNMi views in the context of the selected message. Information embedded in each message supports this cross-navigation:

- The `nnmi.server.name` and `nnmi.server.port` custom message attributes in the message identify the NNMi management server.
- The `nnmi.incident.uuid` custom message attribute identifies the incident in the NNMi database.
- The original trap source appears in the Object column of the HPOM message browser and in the `nnm.source.name` custom message attribute.

Configuring the Agent Implementation

The NNMi integration is installed automatically with HPOM. The NNMi–HPOM integration uses the NNMi northbound integration module and the `nnmopcexport.ovpl` tool, which are part of NNMi 8.12 or higher.

Note: For details about installation and configuration tasks that you must carry out on NNMi management servers, see the *HP NNMi Deployment Guide*.

1. Create an SNMP policy file on the NNMi management server.

   On the NNMi management server, use `nnmopcexport.ovpl` to create an SNMP Trap policy file (`NNMi_policy.dat`), which you can then transfer to the HPOM management server. For information about how to create an SNMP Trap policy file, see the *HP NNMi Deployment Guide*.
2. Enable HPOM to receive messages from NNMi.

   To enable HPOM to receive messages from NNMi, perform the following steps on the HPOM management server:

a. Add a node for the NNMi management server and install an agent on the node.

   For the prerequisites and installation instructions for the HP Operations agent, see the "HP Operations Agent Installation on HPOM Managed Nodes" on page 34.

b. Import the `NNMi_policy.dat` file into HPOM by running the following command:

   ```
   # opcpolicy -add NNMi_policy.dat
   ```

   Note: The `opcpolicy -add` command creates version 1.0 of this policy. If a policy with the name `NNMi Management Events` already exists in the HPOM database, the command fails.

   If you want to keep the old version, rename it before running the `opcpolicy -add` command. If you do not want to keep the old policy, delete it before importing the new one. You can also use the `-replace` option to replace an already existing policy if you do not mind losing the changes, otherwise rename the existing policy and then upload the new one.

c. Deploy the `NNMi Management Events` policy to the NNMi managed node.

d. Add an external node to catch all forwarded NNMi incidents.

   You can set up one external node to catch all forwarded NNMi incidents, eliminating the need to configure each system in HPOM as a separate managed node. For initial testing, set the node filter to `<*>.<*>.<*>.<*>` (for an IP filter) or `<*>` (for a name filter). After you validate the integration, restrict the external node filter to match your network.

Note: If you do not set up an external node for the NNMi incident source nodes, then all incidents forwarded from the NNMi server will be discarded by the HPOM management server.

3. Allocate a port to the HPOM agent on the NNMi management server.

   On the NNMi management server, allocate a custom port number to the HPOM agent to enable the agent to receive SNMP traps from NNMi. For more information about setting up the HPOM agent to receive SNMP traps from NNMi, see the *HP NNMi Deployment Guide*.

4. Configure NNMi incident forwarding to HPOM.

   On the NNMi management server, configure NNMi to forward incidents to HPOM. For more information about setting up incident forwarding in NNMi, see the *HP NNMi Deployment Guide*.

5. *Optional*. On the HPOM management server, add custom message attributes for NNMi incidents to the message browser.

   a. In the browser, right-click any column heading, and then click **Customize Message Browser Columns**.

   b. In the Custom tab, select from the Available Custom Message Attributes, and then click **OK**.

- Most of the custom message attributes for NNMi incidents begin with the text `nnm`.
- The most interesting attributes for NNMi incidents are as follows:

  `nnm.name`

  `nnm.server.name`

c. *Optional*. To change the order in which the custom message attributes appear in the messages browser, drag a column heading to the new location.

6. *Optional*. On the HPOM management server, install additional NNMi tools.

   For details, see "Installing Additional NNMi Tools" on page 553.

NNMi Integration: Web Services Implementation

The NNMi–HPOM integration uses a web services-based integration module to forward incidents automatically from NNMi into the message browser in HPOM server installations. You can also configure filters that limit the criteria under which NNMi forwards incidents to HPOM. The integration synchronizes incidents between NNMi and HPOM. It also provides easy access to the NNMi console and NNMi forms, views, and tools from within HPOM.

The forwarded incidents appear in the HPOM message browser. These messages in the HPOM browser are associated with the original incidents reported in NNMi.

Configuring the Web Services Implementation

The NNMi–HPOM integration is installed automatically with the HPOM installation. HP Incident Web Service (IWS), a prerequisite for the integration, is also an integral part of the HPOM installation.

Note: For details about installation and configuration tasks that you must carry out on NNMi management servers, see the *HP NNMi Deployment Guide*.

1. On the NNMi management server, perform the following configuration steps:
   a. Configure NNMi incident forwarding to HPOM.
   b. Customize the integration.

      For details, see the *HP NNMi Deployment Guide*.
2. In HPOM, create a managed node for each NNMi node that will be named as a source node in the NNMi incidents that are forwarded to this HPOM management server. Also create a managed node for each NNMi management server that will forward incidents to this HPOM management server.

   Alternatively, you can create one external node to catch all forwarded NNMi incidents. For more information about configuring external nodes using the command-line interface, see the *opcnode (1m)* manual page.

Note: Make sure that the NNMi nodes, from which the corresponding NNMi incidents originated, are configured in the HPOM database. If you do not set up these NNMi nodes in the HPOM database, then all incidents forwarded from the NNMi server will be discarded by the HPOM management server.

3. In HPOM, add the custom message attributes for NNMi incidents to the active messages browser, as follows:
   a. In the browser, right-click any column heading, and then click **Customize Message Browser Columns**.
   b. On the Custom tab, select from the Available Custom Message Attributes, and then click **OK**.
      - The custom message attributes for NNMi incidents begin with the prefix `nnm`.
      - The most useful custom message attributes for NNMi incidents are as follows:
        - `nnm.assignedTo`
        - `nnm.category`
        - `nnm.emittingNode.name`
        - `nnm.source.name`
   c. *Optional*. To change the order in which the custom message attributes appear in the messages browser, drag a message-attributes column heading to the new location.
4. *Optional*. On the HPOM system, install additional NNMi tools.

For details, see "Installing Additional NNMi Tools" on page 553.

Synchronization of Incident Updates

When configured to do so, NNMi forwards incidents to one or more HPOM servers. NNMi will acknowledge or unacknowledge an incident to one or more HPOM installations if that incident's life-cycle state changes to or from closed, respectively. Updates to these forwarded incidents are sent from the HPOM server back to the NNMi server to synchronize the life-cycle state of the incident.

Changes to the incident life-cycle state are synchronized between NNMi and HPOM as shown in Table 41.

Table 41: Synchronization of Incident Life-Cycle State Changes

Trigger	Result
Message is acknowledged in HPOM.	Corresponding NNMi incident's life-cycle state is set to Closed.
Message is unacknowledged in HPOM.	Corresponding NNMi incident's life-cycle state is set to Registered.

Synchronization of Incident Life-Cycle State Changes, continued

Trigger	Result
Incident's lifecycle state is set to Closed in NNMi.	Corresponding HPOM message is acknowledged.
Incident's lifecycle state is changed in NNMi from Closed to any other state.	Corresponding HPOM message is unacknowledged.

NNMi Tools

HPOM enables you to launch the NNMi console showing the original incident. You can launch the console in the context of an incident forwarded from NNMi or in the context of an NNMi node that is set up as a managed node in HPOM.

Note: For the NNMi *agent* implementation, it is mandatory to deploy an HPOM agent to the NNMi management server. The NNMi *web services* integration does not require the deployment of the HPOM agent to the NNMi management server.

Each NNMi incident has a unique identifier. Even where HPOM is consolidating NNMi incidents across multiple NNMi server installations, you can trace a particular incident back to its origin in NNMi and investigate it.

The integration enables you to access NNMi forms, views, and tools from within HPOM. Note that you must configure the integrated NNMi tools before you can use them. For more information about configuring NNMi tools in HPOM, see "Installing Additional NNMi Tools" on page 553.

The NNMi integration with HPOM provides the following tool groups, each of which is described in more detail in the tables that follow:

- `NNMi/By Incident`:

  Tools in the `NNMi/By Incident` group require an incident (or message) context to run in. All the information required (incident identifier, source NNMi server name, and port number) is contained in the message forwarded to the HPOM message browser. For more information about the tools in the `NNMi/By Incident` group, see Table 42 .

- `NNMi/By Node (<short hostname>)`:

  Tools in the `NNMi/By Node` group require a node context to run them. For more information about the tools in the `NNMi/By Node` group, see Table 43.

- `NNMi/General (<short hostname>)`:

Tools in the `NNMi/General` group are for the use of general NNMi functions, such as starting the NNMi console, looking at open incidents, or checking the status of NNMi processes and services. No context

is needed to run these tools. For more information about the tools in the NNMi/General group, see Table 44.

- NNMi Int-Admin:

  The NNMi Int-Admin group contains a tool, Create Server Apps, that you use to install additional NNMi tools (those in groups By Node and General) for a specific NNMi server from the HPOM console.

  Before you can use the tools in the By Node and General groups, you need to install them. The installation procedure requires you to specify the NNMi hostname and a port number. For more information about installing NNMi tools, see "Installing Additional NNMi Tools" on page 553.

Table 42 lists the NNMi tools included in the By Incident group created during the integration procedure.

Table 42: Tools in the By Incident Group

Tool	Action Performed
Incident Form	Launches an incident form corresponding to the selected message in a web browser.
Layer 2 Neighbors	Launches a troubleshooting view in a web browser, showing the layer-2 neighbors of the node from which the corresponding NNMi incident originated.
Layer 3 Neighbors	Launches a troubleshooting view in a web browser, showing the layer-3 neighbors of the node from which the corresponding NNMi incident originated.
Node Form	Launches a node form in a web browser, showing the NNMi setup information for the node from which the corresponding NNMi incident originated.

Table 43 lists the NNMi tools included in the By Node group created during the integration procedure.

Table 43: Tools in the By Node Group

Tool	Action Performed
Comm. Configuration	Launches the real-time results of the ICMP and SNMP configuration report in a web browser, showing the communication configuration of a selected node.
Configuration Poll	Launches the configuration poll of a selected node, showing the real-time results of a node's configuration in a web browser.
Layer 2 Neighbors	Launches a troubleshooting view in a web browser, showing the layer

Tools in the By Node Group, continued

Tool	Action Performed
	2 neighbors of a selected node.
Layer 3 Neighbors	Launches a troubleshooting view in a web browser, showing the layer 3 neighbors of a selected node.
Node Form	Launches a node form in a web browser, showing details about the selected node for troubleshooting purposes.
Ping	Launches the `ping` command and shows the real-time results of the ping from the NNMi management server to a selected node in a web browser.
Status Poll	Launches the real-time check and results of a node's status in a web browser.
Traceroute	Launches the real-time results of the Trace Route command in a web browser.

Table 44 lists the NNMi tools included in the `General` group created during the integration procedure.

Table 44: Tools in the General Group

Tool	Action Performed
My Incidents	Launches the My Open Incidents view in a web browser.
NNMi Console	Launches the NNMi console.
NNMi Status	Launches a report of the current status of all NNMi processes and services in a web browser.
Open RC Incidents	Launches the Open Root Cause Incidents view in a web browser.
Sign In/Out Audit Log	Displays the current configuration for a node in a web browser (tracks log-in and log-out activity for each NNMi user account).

Installing Additional NNMi Tools

You can also install additional NNMi-specific tools in the main NNMi tools group. The additional tools are placed in the following groups:

- `General`:

  For more information about the `General` tools group and the NNMi-specific tools it contains, see Table 44

- `By Node`:

  For more information about the `By Node` tools group and the NNMi-specific tools it contains, see Table 43.

- `By Incident`:

  For more information about the `By Incident` tools group and the NNMi-specific tools it contains, see Table 42.

You can install the additional tools for a specific NNMi management server using one of the following methods:

- NNMi application-installation script:

  For more information about the NNMi Application Installation script see "NNMi Application Installation Script" on page 554.

- `Create Server Apps` tool:

  For more information about the `Create Server Apps` tool, which you can start in the HPOM console, see "Create-Server-Applications Tool" on page 556

NNMi Application Installation Script

HPOM provides a dedicated NNMi Application Installation script that enables you to install additional tools. You can execute the script with or without specifying the server parameters. For example, if you want to choose your own short hostname for labeling the tools group you are installing, execute the script without entering the server parameters.

Installing NNMi Application with Server Parameters

To run the NNMi Application Installation script by specifying the server parameters, use the `create_nnm_appls.sh` script as follows:

```
# /opt/OV/contrib/OpC/NNMi-Appls/create_nnm_appls.sh <fully qualified hostname>
<server port number>
```

This script specifies the fully qualified hostname and the server port number.

The tools group is created in the main NNMi group, and is identified by the short hostname. The short hostname is created automatically using the first part of the fully qualified hostname (truncated at the first dot).

Installing NNMi Application without Server Parameters

To run the NNMi Application Installation script without specifying the server parameters, perform the following steps:

1. Create NNMi applications by running the following script:

   ```
   # /opt/OV/contrib/OpC/NNMi-Appls/create_nnm_appls.sh
   ```

2. Specify the NNMi server by responding to prompts for information, for example: the fully qualified hostname of the NNMi server system, a short hostname, and the port number to use for connections and communication, for example:

   ```
   # create_nnm_appls.sh
   Full qualified name of the NNMi system:
   nnmsv1.example.com

   Short name of the NNMi system [nnmsv1]:
   nnmsv1

   Is the NNM system a NNMi 9 system using HTTPS (y/n)[yes]:
   y

   Port to access the NNMi system [8004]:
   8004
   ==================================================
   NNMi 9 (HTTPS):  yes
   System Name: nnmsv1.example.com
   Short Name: nnmsv1
   Port:        8004
   Used Locale: en_US.UTF-8
   ==================================================
   Are these parameters correct?
   Press [ENTER] to proceed or [^C] to cancel.
   Done
   ```

3. Verify that the information you entered is correct, and press **ENTER** to install the tools.

   The new tools group created in the main NNMi group is identified by the short hostname which you provided in response to the prompt during installation. You can move the group to a more suitable place if desired.

4. Assign the created tools or tools groups to the appropriate operators.

   Operators might be required to reload the configuration if they are working in the user interfaces, when the change takes place. To reload the configuration to a running interface, use the feature File -> Reload Configuration.

Create-Server-Applications Tool

You can install the additional NNMI tools directly from within the HPOM console by using the `Create Server Apps` tool. The new tools group is created in the main `NNMi` tools group and is identified by the short hostname. The short hostname is created automatically using the first part of the fully qualified hostname (truncated at the first dot).

Creating NNMi Tools from the HPOM Console

To install the additional NNMi tools using the `Create Server Apps` tool, perform the following steps:

1. In the HPOM console, double-click **Tools**, and then double-click **NNMi Int-Admin**.
2. Right-click **Create Server Apps** and then select **Start Customized**.

   Note: If you try to start the `Create Server Apps` tool by double-clicking, an error is reported in the output window.

3. In the dialog box that opens, select the node on which you want to run the `Create Server Apps` tool. Click **Next** to continue.
4. Enter additional information needed to run the tool.

   In the Additional Parameters field, enter the fully qualified hostname of the NNMi server and its port number. Click **Finish** to end the installation of the additional NNMi tools.
5. Select **File -> Reload Configuration**.

   The Configuration Status window opens. Click **OK** when the reload is done.

Launching NNMi Tools from the HPOM Console

The tools listed in the section "NNMi Tools" on page 551 can only be run after you install them. For more information about installing NNMi tools, see "Installing Additional NNMi Tools" on page 553. For more information about using the tools you install, see the follow sections:

- "Launching an NNMi Incident Form" on page 556
- "Launching the NNMi Console" on page 557

Launching an NNMi Incident Form

To launch an NNMi Incident Form from within the HPOM console, perform the following steps:

1. Browse the list of messages in the HPOM Message Browser and locate a message forwarded from NNMi.
2. Right click the NNMi message, and select the following menu option: **Start -> NNMi -> By Incident -> Incident Form**

The first time you run the tool, the log-in screen for NNMi opens and prompts for logon credentials.

3. Enter a user name and password and click **Sign-In**. The NNMi Incident Form opens.

Launching the NNMi Console

To launch the NNMi console from the HPOM user interface, perform the following steps:

1. Select the following menu option: **Tools -> NNMi**
2. Select the following tool: **General (*<host>*)**, where *<host>* is the short hostname of the NNMi server that you want to log on to.
3. Select the following tool: **NNMi Console**

When NNMi displays the log-in screen, type the User Name and Password and then click **Sign-In** to open the NNMi console.

Web Browser Settings

You must configure the web browser settings for the console according to operating-system platform, as follows:

- Windows platforms:

  Configure the console to use either an external web browser or the Internet Explorer ActiveX control.
- Other platforms:

  Configure the console to use an external web browser.

Note: Choose a web browser that is supported by NNMi version you use.

Modifying Web-Browser Settings

To check or change the web-browser settings for the console, perform the following steps:

1. In the tool bar, click **Edit**, then click **Preferences**.
2. Click the Web Browser tab in the Preferences dialog box.
3. Select the browser settings as appropriate for your platform.
4. Click **OK**.

Chapter 13: Notification Services and Trouble-Ticket Systems

In this Chapter

This chapter explains what you need to consider when configuring a link between HPOM and an external notification service or an external trouble-ticket system. The information in this chapter explains how to write scripts and programs to automatically call an external notification service or an external trouble-ticket system when a message is received on the management server. It also describes the high-level steps used to integrate an external notification service or trouble-ticket system into HPOM. Finally, this chapter describes the parameters provided by HPOM to call a notification service, and to forward a message to a trouble-ticket system.

The information in this section covers the following topics:

Notification Services and Trouble-Ticket Systems

You can configure HPOM to automatically call an external notification service or an external trouble-ticket system when a message is received on the management server. You can set up programs and scripts to notify users by modem, telephone, or email. You can also send event-specific details to a trouble-ticket system you have predefined.

- Notification service:

  A notification service can be any form of communication that is used to inform an operator of a very important event. For example, you could use a pager, send a Short Messaging Service (SMS), or an email. HPOM allows you to set up different notification mechanisms for each of your operators. In addition, you can schedule your external notification services according to a timetable.

- Trouble-ticket system:

  Trouble-ticket systems are used to document, track, and help resolve reported problems.

- HP service desk:

HP Service Desk enables you to manage all aspects of your business processes. Service Desk is tightly integrated with HPOM, which means you can configure HPOM to forward either *all* events or specific *individual* events to Service Desk. The integration enables event information to be mapped to a Service Desk incident. The first time an event is sent, an incident is created in Service Desk. Service Desk then becomes the owner of the event. The mapping process in Service Desk defines which event attributes will be imported into the Incident fields. For more information about the integration, see the Service Desk product information.

Note: In case you need to pause or resume the trouble ticket interface and the notification services (for example, when you want to perform maintenance on agents), use the `-pause` or `-resume` option with the `opctt` command.

Scripts and Programs

HPOM enables you to write your own script or program that calls an external interface such as a notification service or a trouble-ticket system. The script serves as a link between HPOM and the notification service or trouble-ticket system.

HPOM provides an example script that shows you how to call and make use of an external notification service or trouble-ticket system. The following script sends an email to all operators responsible for the message that is configured to call the service or trouble-ticket system:

```
/opt/OV/bin/OpC/extern_intf/ttns_mail.sh
```

Guidelines for Writing Scripts and Programs

When writing your script or program, note the following important guidelines:

- Default directory:

  For scripts and programs calling external interfaces, you can use the following default directory provided by HPOM:

  ```
  /opt/OV/bin/OpC/extern_intf
  ```

  Caution: If you place your scripts and programs in this directory, they will be erased when you deinstall HPOM.

- Shell scripts:

  Scripts are executed under the account of the user who started the HPOM server processes. In most cases this is the user root.

  If your script is a shell script, the first line must contain a statement such as the following:

  ```
  #!/usr/bin/sh
  ```

This statement ensures that the shell for which your script is designed is used during execution, and not the shell of the user who executes the script.

Caution: If the first line of your shell script does not contain this statement, the execution of your script or program may fail.

- Default Parameters:

  HPOM sends its own message parameters to the external interface. You may *not* use a command that requires additional parameters. For a list of the parameters provided by HPOM, see "Parameters for Notification Services and Trouble-Ticket Systems" on page 562.

Integration of Notification Services and Trouble-Ticket Systems

This section explains how to integrate an external notification service or trouble-ticket system with HPOM. The high-level steps described in this section provide you with an overview of the following configuration tasks:

- "Configuring Notification Services" on page 560
- "Configuring Trouble-Ticket Systems" on page 561

Configuring Notification Services

To configure an external notification service for integration with HPOM, perform the following steps:

1. Set up the notification service.

   To set up a notification service, do the following:

   a. Write a script or program that calls the notification service.

      For details, see "Guidelines for Writing Scripts and Programs" on page 559.

   b. Set up a notification method by using the `opcnotiservice` command. For more information about command options and parameters, see the manual page for the *opcnotiservice(1m)* command.

2. Set the notification schedule.

   To set a notification schedule, use the `opcnotischedule` command and configure the following values:

   a. The schedule that your external notification services must adhere to, according to a defined timetable.

   b. The notification services used and at what time during the week.

For example, you could schedule a phone call at work during working hours, and a phone call at home during evenings and weekends. For more information about command options and parameters, see the manual page for the *opcnotischedule(1m)* command.

3. Set external notification for a message condition:

   Configure messages to be forwarded to the external notification service according to the schedule you set. Define which messages send external notifications by setting a switch in the corresponding condition in the policy.

   Tip: Instead of modifying each condition separately, you can set up a global flexible-management policy for service hours and scheduled outages. The global policy defines which messages are forwarded to the notification service. For more information, see the *Administration UI Help*.

Configuring Trouble-Ticket Systems

To configure a trouble-ticket system for integration with HPOM, perform the following steps:

1. Set up the trouble-ticket system.
   a. Write a script or program that calls the trouble-ticket system.

      For details, see "Guidelines for Writing Scripts and Programs" on page 559.

   b. Set up a trouble-ticket call by using the `opctt` command with the `-enable` parameter, as follows:

      `/opt/OV/bin/OpC/opctt -enable /opt/OV/bin/OpC/extern_intf/<ttns_script>.sh`

      Where `<ttns_script>.sh` is a script or a program that calls the trouble-ticket system, for example, `ttns_mail.sh`.

      For more information about the `opctt` command and permitted parameters, see the *opctt(1m)* manual page.

2. Forward messages to a trouble-ticket system.

   Configure messages to be forwarded to the trouble-ticket system. For example, define which messages are forwarded to the trouble-ticket system by setting a switch in the corresponding condition in the policy.

   Tip: Instead of modifying each condition separately, you can also set up a global flexible-management policy for service hours and scheduled outages to define which messages are forwarded to the trouble-ticket system. For more information, see the *Administration UI Help*.

You cannot schedule sending event-specific details to a predefined trouble-ticket system.

Parameters for Notification Services and Trouble-Ticket Systems

Table 45 lists the parameters that you can use when writing scripts that call a notification service or a trouble-ticket system, or inform operators who are responsible for messages that include a call to a trouble-ticket system or a notification service. The parameters have a specific order as illustrated in Table 45.

Table 45: Permitted Parameters for Notification Services and Trouble-Ticket Systems

Parameter Number	Parameter	Description	Example
1	message_id	Unique message number	`c1c79228-ae12-71d6 -1a8f-0f887ebe0000`
2	source_node	Message node name	`nodename.hp.com`
3	source_node_type	Node type	`HP 9000 PA-RISC`
4	date_created	Date (`mm/dd/yyyy`) on which the message was received on the managed node in the time zone (system-specific `TZ` variable) of the management server.	`08/02/2002`
5	time_created	Time (`hh:mm:ss`) at which the message was received on the managed node. This time uses a 24-hour clock in the time zone (system-specific `TZ` variable) of the management server.	`16:22:04`
6	date_received	Date (`mm/dd/yyyy`) on which the message was received on the management server in the time zone (system-specific `TZ` variable) of the management server.	`08/02/2008`
7	time_received	Time (`hh:mm:ss`) at which the message was received on the management server. This time	`16:22:05`

Permitted Parameters for Notification Services and Trouble-Ticket Systems, continued

Parameter Number	Parameter	Description	Example
		uses a 24-hour clock in the time zone (system-specific `TZ` variable) of the management server.	
8	application	Application name	`/bin/su(1) Switch User`
9	message_group	Message group	`Security`
10	object	Object name	`root`
11	severity	Message severity	`unknown`, `normal`, `warning`, `minor`, `major` or `critical`
12	operators	List of responsible HPOM operators. Names are separated with a single space.	`opc_op Bill John`
13	message_text	Message text. Note that text is *not* enclosed in quotes ("").	`Succeeded switch user to root by charlie`
14	instruction	Instructions (empty string if not available). The instructions are passed without quotation marks (""), backslashes (\), or other characters that might be interpreted by a UNIX shell.	`This is the instruction text for the appropriate message condition. It is available for the operator when a message matching this condition displays in the Message Browser.`
15	cma	Custom message attributes (empty string if not available). Multiple *`name=value`* pairs are separated with two semi-colons (`;;`).	`Customer=Hewlett-Packard;;Country=United States of America`
16	service_name	Service name	`sample_svc1`
17	message_type	Message type	`succeeded_su`
18	message_key	Message key	`appl_`

Permitted Parameters for Notification Services and Trouble-Ticket Systems, continued

Parameter Number	Parameter	Description	Example
			`status:myapp1:disk:nodename.hp.com`
19	supp_dup_msgs	Number of suppressed duplicate messages. The number is `0` unless at least one of the following parameters has been set to `TRUE` using the `ovconfchg` command line tool: • `OPC_NOTIF_WHEN_DUPLICATE` Passes duplicate messages to the notification interfaces with a 19th parameter containing the duplicate counter. The counter is zero if it is the first message or this feature is not switched on. • `OPC_TT_WHEN_DUPLICATE` Passes messages to trouble-ticket systems even if they are duplicates of other messages.	`14`
20 *(optional)*	forward_manager	Name of the HP Operations management server that forwarded a message.	`fwdmgr.hp.com`

Part V: Security

HP Operations Manager (HPOM) security is impacted by the wider environment that HPOM monitors. Administrators can improve security practices by taking into consideration the wider context of HPOM functioning.

Smart card authentication is a method of enhancing HPOM security, through the use of certificate technologies to authorize users.

For detailed information on HPOM security and smart card authentication, see:

- "HPOM Security" on page 566
- "Smart Card Authentication" on page 611

Chapter 14: HPOM Security

In this Chapter

This chapter explains the security measures you can investigate and implement in the wider context of HP Operations Manager (HPOM) for example: system security, network security, HPOM security, log-in authentication, and best practices for system audits. The information in this chapter is organized into the following topic areas:

Security Overview

The security of your HPOM system involves much more than the configuration of the HPOM software; it also requires attention to security matters in the wider environment that is monitoring. In particular, you should investigate how you can improve security practices in the following areas:

- System security:

  Enable the HP Operations management server and managed node to run on a “trusted” system.

  For details, see "System Security" on page 567.

- Network security:

  Protect data that is exchanged between the management server and the managed node.

  For details, see "Network Security" on page 568.

- HPOM security:

  Investigate security-related aspects of application setup and execution, operator-initiated actions, and HPOM auditing.

  For details, see "HPOM Security" on page 570 and "HPOM Audits" on page 600.

Note: To find out how HPOM behaves in an environment protected by firewalls, see the *HPOM Firewall Concepts and Configuration Guide*.

System Security

This section describes how HPOM behaves in trusted system environments.

Note: Before installing and running HPOM on any system, you must ensure that the system-level security measures comply with your organization's policies regarding system security. To learn about system-level security policies, see the product documentation for the relevant operating systems as well as your specific company guidelines.

Guidelines for System Security

A secure or "trusted" system uses a number of techniques to improve security at the system level. Many different system-related security policies exist, ranging from standards with industry-wide recognition such as the Controlled Access Protection (C2) system (developed by the U. S. Department of Defense) to standards that are established and used internally in IT departments within enterprises.

Note: Installing and running HPOM in a C2-secure environment has not yet been certified.

Standards for system security vary in stringency and apply a variety of techniques, including the following:

- Authentication:

  System security standards may impose strict password and user-authentication methods for user-logon procedures. HPOM supports the authentication module (PAM) for the authentication of users during the Java GUI logon sequence. PAM enables multiple authentication technologies to be added without changing any of the logon services, thereby preserving existing system environments. For more information about PAM authentication, see "PAM Authentication" on page 573.

  When implementing system-related security standards, be aware that password aging and changing can lead to problems with application startup if any passwords have been hard coded in HPOM.

- Auditing:

  System security standards may require regular auditing of networking, shared memory, file systems, and so on. HPOM enables the auditing of any kind of user interaction within HPOM. For further details, see "HPOM Audits" on page 600.

- Terminal access and remote access:

  System security standards may include measures to control access to terminals. If the system security policy disallows root logon through the network, HPOM agents must be installed manually.

- File access:

  System security standards may include measures to manage access to files. Some policies recommend the use of access control lists (ACLs). When maintaining the system security standard on a system running HPOM, be aware that HPOM does not use ACLs. HPOM imposes strict file access permissions, and protects important files either by encrypting them or by using digital signatures.

Network Security

In HPOM, network security is designed to improve the security of connections between processes. These secure process connections can be within a network, across multiple networks, or through routers or other restrictive devices.

For example, you could limit access to a network or a section of a network by restricting the set of nodes (with or without HPOM agents running on them) that are allowed to communicate with the management server across restrictive routers or even a packet-filtering firewall. HPOM provides robust security regardless of whether the server or the network of managed nodes are inside or outside the firewall. A management server outside your firewall can manage a network of nodes inside your firewall. Conversely, a management server inside your firewall can manage nodes outside your firewall.

One way of limiting access to a network, and consequently improving the network's inherent security, is to restrict all connections between HPOM processes on the management server and a managed node to a specific range of ports. To simplify matters, HPOM sets the default value on the managed node to "No security," and enables you to select the security configuration node by node. In this way, you can change the security of a given node, depending, for example, on whether there is a need for the node to communicate across a firewall or through a restricted router.

HTTPS Security

HTTPS 1.1 based communication is the communication technology used for HP Software products and enables applications to exchange data between heterogeneous systems.

HTTPS communication uses the Secure Socket Layer (SSL) protocol to validate access to data and secure the exchange of data. With businesses sending and receiving transactions across the Internet and private intranets, security and authentication assume an especially important role.

HTTPS communication meets this goal through the implementation of established industry standards. The combination of HTTPS (HTTP with SSL encryption) and authentication ensure data integrity and privacy:

- Data compression:

  By default, data is compressed, ensuring that data is not transmitted in clear text format, even for non-SSL connections.

- Single Port Entry:

  All remote messages arrive through the Communication Broker, providing a single port entry to the node.

- Custom Port Range:

  You may specify a restricted bind port range for use in configuring firewalls.

- Firewalls and proxies:

  When sending messages, files, or objects, you may configure one or more standard HTTP proxies to cross a firewall or reach a remote system.

For further information about HTTPS security in HPOM, see the HP Operations agent documentation.

Secure Shell

The HPOM agent software can alternatively be installed using the Secure Shell (SSH) installation method. For details, see "Secure Shell Installation" on page 42.

Secure Shell (SSH) is a UNIX program that can be used to log on to and execute commands on a remote computer. SSH is intended to replace `rlogin` and `rsh` and provide secure encrypted communications between two untrusted hosts over an insecure network. X11 connections and arbitrary TCP/IP ports can also be forwarded over the secure channel. The SSH provides a number of security features, such as:

- Port forwarding:

  All communication between two systems is conducted between well-known ports, thereby creating a virtual encrypted communication channel.

- RSA authentication:

  All logons, even those without a password, use RSA authentication.

- Public-key encryption:

  All traffic between systems is secured with public-key encryption.

HPOM Agent Installation Using Secure Shell

The secure-shell (SSH) installation method provides enhanced security for installations that are performed using insecure connections (for example, over the Internet).

The agent-installation process uses the secure-copy (SCP) tool to transfer files between source and target hosts, and remote commands are executed using the command-execution facility built into SSH.

The emphasis on increased security helps to reduce the risk of people or programs eavesdropping on (or tampering with) communications between systems.

The HPOM installation process works with any configuration already established on the management server, regardless of security features used, as long as you set up an automatic logon for user `root` on the managed node. For example, you can set up an automatic logon on the managed node by establishing a logon based on RSA authentication, which does not require a password. For more information, see "Installing HP Operations Agent Software by Using SSH" on page 42.

HPOM Security

As an HPOM administrator, you need to carefully think through the security implications of your HPOM configurations. For example, managed nodes allow only those management servers that they recognize as action-allowed managers to execute operator-initiated actions.

Access to HPOM

Only registered HPOM users can access the Java GUI. By default, the users `opc_adm` and `opc_op` are available.

HPOM Operator Passwords

As an HPOM administrator, you can change the passwords defined for HPOM operators. However, you cannot see new passwords set by operators (the characters appear as asterisks). By default, operators can change their own passwords.

Preventing Operators from Changing Passwords

To prevent HPOM all operators from changing their logon password, perform the following steps:

1. Open the `opcop` file, which you can find in the following location:

   `/etc/opt/OV/share/conf/OpC/mgmt_sv/appl/registration/C/opc_op/opcop`
2. Add the following lines to the file:

   ```
   Action “Change Password”
   {
   }
   ```
3. Save the changes.

Java GUI Permissions

The HPOM Java-based operator GUI communicates with the HP Operations management server through port 2531. The `inetd` (on HP-UX and Solaris) or `xinetd` (on Linux) monitors port 2531 and

starts the process `/opt/OV/bin/OpC/opcuiwww` when request is received for the service `ito-e-gui`.

By default, the HP Operations management server accepts connections from any client. On the management server, you can restrict client access to specific systems as follows:

- On HP-UX:

  Edit the `/var/adm/inetd.sec` file. Remember to specify the systems permitted to access the service `ito-e-gui`.

- On Solaris:

  Enable the TCP Wrappers for the `ito-e-gui` service. After doing this, the access to it is controlled with the `/etc/hosts.allow` and `/etc/hosts.deny` files.

- On Linux:

  Edit the `/etc/xinetd.d/ito-e-gui` file.

For more information about making the connection between the Java GUI client and the HPOM management server more secure, for example, by using HTTPS and an alternative port number, see "Secure HTTPS-based Communication" on page 324.

Secure File Upload

Uploaded files present a significant risk to applications. Malicious file uploads lead to a complete system takeover, an overloaded file system or database, forwarding attacks to back-end systems, and defacement.

To avoid malicious file uploads the HPOM server should validate mime type and size limit. These settings are configured in `/opt/OV/OMU/adminUI/conf/cocoon.properties` file as shown below:

1. To allow specific mime types, uncomment the property "`midas.mimetypes.allowed`". Add the required mime types separated with a comma. The end of line should not have full stop.

   Example:

   To allow only plain text and XML files:

   ```
   midas.mimetypes.allowed=text/plain, application/xml
   ```

   By default this property is commented, allowing all mime types to be uploaded.

2. To set the maximum allowable size of a file, set the property “`midas.upload.maxsize`”. The value is set in bytes.

   Example:

   To allow 50000 Bytes at max for a file:

   ```
   midas.upload.maxsize=50000
   ```

   By default the maximum allowable size of a file is kept "unlimited" as shown below:

`midas.upload.maxsize=`

You can comment out the property by using # preceding "`midas.upload.maxsize`".

Database Security

Database security is controlled by the operating system and by the database itself. Users must have an operating-system logon to be able to access the database either remotely or locally. After a user logs on, the database security mechanisms assume control of any requests to access the database and database tables.

For more information about database security, see the product documentation supplied with the database software.

Starting Applications

Applications run under the account (user and password) specified by the administrator during application configuration. The HPOM action agent uses the information in this account before executing an application. The action agent switches to the user specified and then uses the name and password stored in the application request to start the application.

User Root

If the user account under which the HPOM agents are running has been switched to a user other than root, you have to carry out additional configuration steps. For more information about command options and parameters, see the *ovswitchuser(1)* manual page .

Password Aging

Password aging is a standard security feature that requires passwords to expire automatically, for example, if one of the following rules applies:

- Time:

  A specified period of time has passed since the password was last changed.
- Date:

  A specified date has been reached without the password being changed.
- Number:

  A specified number of unsuccessful logon attempts have been made by the user associated with a particular password.

Password aging can compromise the startup and execution of applications. For example, if password aging is enabled, application startup failures can occur if the user account that a given application uses is temporarily inaccessible. You can reduce the occurrence of application-startup failures by

configuring the HPOM plug-in interface for the PAM authentication module, which enables third-party authentication methods to be used while preserving existing system environments. For more information, see "PAM Authentication" on page 573.

PAM Authentication

You can use a Plug-in Authentication Module (PAM) to retrieve and check user names and password information. The user information is saved in a central repository to which the PAM module has access. To set up PAM for authentication, use the `ovconfchg` command on the HP Operations management server. For more information about the `ovconfchg` command, see the *ovconfchg(1m)* manual page.

The HPOM user model requires users (humans or programs) to log on to the HP Operations management server before being able to use any further functionality. This applies mainly to the Java GUI, but also to some of the application-programming interfaces (API) and the command-line interface (CLI) of the HP Operations management server.

The logon procedure includes the following checks:

- Authenticate the user and verify access permission.
- Determine the user's capabilities.

HPOM enables you to use PAM for authentication instead of the built-in authentication mechanism. Using PAM has the following major advantages:

- Common user database

  PAM shares a common user database with the operating system and other applications. This enables the setup and management of user accounts and passwords in one place.
- High security

  PAM authentication enables the implementation of high security policies including: stronger encryption, password aging, account expiration, and so on.

Note: PAM-specific security measures apply only to the user-authentication process. The HPOM user accounts must still exist to determine the user's capabilities.

The following restrictions apply to PAM user authentication with HPOM:

- Account or session management

  HPOM PAM does not support either the management of PAM accounts or PAM-authenticated sessions. HPOM uses PAM only for authentication.
- Account setup

  Account setup and management (including the password update) must be performed by using external tools with regards to the used PAM mechanism. For example, if the UNIX `passwd` PAM

module is used, then you must use standard UNIX commands to manage user accounts and passwords on the operating-system (OS) level.

The HPOM password tool updates only user passwords in the HPOM database. PAM does not consider passwords in the HPOM database for authentication purposes. If PAM authentication is enabled, use external tools to modify or set user passwords.

- Password requests

  It is not possible to use authentication stacks which request multiple passwords.

Caution: When using PAM authentication on RHEL 6.x, make sure that you have the `compat-openldap` package version 2.3.43 or higher installed.

Configuring PAM User Authentication

To configure HPOM user authentication to use the PAM module, perform the following steps:

1. Enable PAM user authentication in HPOM. Set the variable `OPC_USE_PAM_AUTH` to `TRUE` as follows:

   ```
   /opt/OV/bin/ovconfchg -ovrg server -ns opc -set OPC_USE_PAM_AUTH TRUE
   ```

   This setting instructs HPOM to use PAM as the authentication mechanism.
2. Configure PAM to route HPOM authentication requests to the desired PAM module as follows:
   - On HP-UX and Solaris

     Add the following entry to the PAM configuration file `pam.conf`:

     ```
     ovo       auth      required      <module>
     ```

     Where the meaning of the parameters stated in the entry is as follows:

`ovo`	HPOM application ID.
`auth`	Module used for authentication only.
`required`	Authentication step must be successful.
`<module>`	Name of the PAM module to be used. Technically, this a shared library that implements the authentication mechanism (for example, UNIX `passwd`, Kerberos, NIS, or LDAP).

     For more information about the contents of the `pam.conf` file, see the *pam.conf(5)* manual page.

   - On RHEL

     Create and edit the PAM configuration file `/etc/pam.d/ovo`.

For examples on how to configure PAM authentication by using different authentication mechanisms, see "Examples of Configuring PAM Authentication" on page 576.

3. *Optional:* Set user-based or module-specific flags. For more information, see the PAM documentation.
4. Create user names and corresponding passwords for the HPOM administrator (`opc_adm`) and each of the HPOM operators. You might have to use external tools, depending on the selected PAM mechanism.
5. Create the remaining HPOM operator accounts in HPOM and assign the required responsibilities. Log on to HPOM as `opc_adm` by using the password specified in the previous step.

Disabling PAM User Authentication

To disable PAM user authentication in HPOM, set the variable `OPC_USE_PAM_AUTH` to `FALSE` as follows:

```
/opt/OV/bin/ovconfchg -ovrg server -ns opc -set OPC_USE_PAM_AUTH FALSE
```

Counting Failed PAM-Authenticated Logons

You can count the number of times the PAM authentication process registers a failed attempt to log on to the Java GUI for each user or operator. The value of that failed-logon counter is stored as a configuration variable in the operator's name space `user.<username>`.

To enable PAM to count the number of failed attempts to log on to HPOM by using the Java GUI, perform the following steps:

1. Enable PAM user authentication by using the `ovconfchg` command as follows:

   ```
   /opt/OV/bin/ovconfchg -ovrg server -ns opc -set OPC_USE_PAM_AUTH TRUE
   ```
2. Set the counter for failed PAM-authenticated logons by using the `ovconfchg` command as follows:

   ```
   /opt/OV/bin/ovconfchg -ovrg server -ns opc -set OPC_USE_PAM_FAILED_LOGIN_
   COUNTER TRUE
   ```

After the third failed logon, the following configuration variables are updated in each `user.<username>` name space. Note that the stated values are only examples:

`FAILED_LOGIN_ATTEMPT_COUNTER=3` (Counter)

`LAST_FAILED_LOGIN_ATTEMPT=1197559311` (Time in seconds since epoch)

`LOGIN_ATTEMPT_DELAY=60` (Delay in seconds)

You can list the current values by using the following command:

```
/opt/OV/bin/ovconfget -ovrg server user.<username>
```

Note: After the third failed logon, all further logons for this user are blocked until `LOGIN_ATTEMPT_DELAY` expires.

It is possible to overwrite the current values of the configuration variables. For example, you can reset counter, time, or delay by using the following commands:

```
/opt/OV/bin/ovconfchg -ovrg server -ns user.<username> -set FAILED_LOGIN_ATTEMPT_
COUNTER 0
```

```
/opt/OV/bin/ovconfchg -ovrg server -ns user.<username> -clear LAST_FAILED_LOGIN_
ATTEMPT -clear LOGIN_ATTEMPT_DELAY
```

If you want both the `FAILED_LOGIN_ATTEMPT_COUNTER` and `LOGIN_ATTEMPT_DELAY` server configuration variables to be reset automatically after each successful PAM-authenticated logon, set the `OPC_USE_PAM_RESET_COUNTER_AND_DELAY` server configuration variable to `TRUE` as follows:

```
/opt/OV/bin/ovconfchg -ovrg server -ns opc -set OPC_USE_PAM_RESET_COUNTER_AND_DELAY
TRUE
```

Examples of Configuring PAM Authentication

You can configure PAM authentication by using different authentication mechanisms. This section contains the examples of using UNIX password, Kerberos, and LDAP authentication mechanisms.

Caution: Before you configure PAM authentication on your system, see the documentation for implementing PAM specific for your operating system.

Using the UNIX Password Authentication

To use the UNIX password authentication, perform the following:

- On HP-UX

  Edit the `/etc/pam.conf` file for the `ovo` module, by adding the following lines:

  ```
  ovo auth required /usr/lib/security/hpux32/libpam_unix.so.1
  ovo account required /usr/lib/security/hpux32/libpam_unix.so.1
  ```

- On Solaris

  Edit the `/etc/pam.conf` file for the `ovo` module, by adding the following lines:

  ```
  ovo   auth     requisite  pam_authtok_get.so.1
  ovo   auth     required   pam_unix_auth.so.1
  ovo   account  required   pam_unix_account.so.1
  ```

- On RHEL

  Create the `/etc/pam.d/ovo` PAM module. Edit the `/etc/pam.d/ovo` file by adding the following lines:

  ```
  #%PAM-1.0
  auth      sufficient   pam_unix.so nullok try_first_pass
  auth      required     pam_deny.so
  account   required     pam_unix.so
  account   required     pam_permit.so
  ```

Using the Kerberos Authentication

To use the Kerberos authentication, perform the following:

- On HP-UX

  Edit the `/etc/pam.conf` file for the `ovo` module, by adding the following lines:

```
ovo   auth     required   libpam_krb5.so.1
ovo   account  required   libpam_krb5.so.1
```

- On Solaris

  Edit the `/etc/pam.conf` file for the `ovo` module, by adding the following lines:

```
#%PAM-1.0
ovo auth requisite pam_authtok_get.so.1
ovo auth required pam_krb5.so.1
ovo account required pam_krb5.so.1
```

- On RHEL

  Create the `ovo` PAM module, `/etc/pam.d/ovo`. Edit the `/etc/pam.d/ovo` file by adding the following lines:

```
#%PAM-1.0
auth        sufficient    pam_krb5.so
auth        required      pam_deny.so
account     required      pam_permit.so
```

Using LDAP Authentication

Make sure that the LDAP client software is installed on your system (for example, LDAP-UX on HP-UX or OpenLDAP for all the platforms). An LDAP client should be configured to point to your LDAP server.

To use LDAP authentication, you can use one of the following files:

- `pam.conf` file, which is specific to the operating system
- `ldap.conf` file, which can be used on platforms that support such configuration

Using the `pam.conf` file:

- On HP-UX

  Edit the `/etc/pam.conf` file for the `ovo` module, by adding the following lines:

```
ovo     auth        required        /usr/lib/security/hpux32/libpam_ldap.so.1
ovo     account     required        /usr/lib/security/hpux32/libpam_ldap.so.1
```

- On Solaris

Edit the /etc/pam.conf file for the ovo module, by adding the following lines:

```
ovo auth        requisite       pam_authtok_get.so.1
ovo auth        required        pam_ldap.so.1
ovo account     required        pam_ldap.so.1
```

- On RHEL

  Create the /etc/pam.d/ovo PAM module. Edit the /etc/pam.d/ovo file by adding the following lines:

```
#%PAM-1.0
auth            sufficient      pam_ldap.so
auth            required        pam_deny.so
account         required        pam_permit.so
```

Using the ldap.conf file:

You can use the config option in the PAM module to configure the ldap.conf file on platforms that support such configuration.

Example of using the ldap.conf file on RHEL:

Create the /etc/pam.d/ovo PAM module and edit the /etc/pam.d/ovo file:

```
%PAM-1.0
auth       sufficient    pam_ldap.so       config=/opt/OV/adm/conf/ovoldap.conf
auth       required      pam_deny.so
account    required      pam_permit.so

cat /opt/OV/adm/conf/ovoldap.conf
pam_login_attribute sAMAccountName
binddn <bind user>
bindpw <password for bind user>
base <base DN>
uri <ldap uri> <ldap uri 2> <ldap uri 3>
```

Configuring PAM User Authentication by Using LDAP, Likewise Open, or Winbind on RHEL with Windows Active Directory

To enable logon of the Windows users to the RHEL systems that host HPOM, you can configure PAM authentication with Windows Active Directory. You can use different authentication mechanisms. This

section contains the examples of using LDAP, Likewise Open, and Winbind.

Using the LDAP Authentication with Windows Active Directory

Use this example to configure PAM authentication by using LDAP. This is required to enable access to the Windows Server 2008 Active Directory from the RHEL 5.x systems. Encrypted communication with LDAP server is not covered with this example.

Make sure that the LDAP client software is installed on your system (for example, OpenLDAP). An LDAP client should be configured to point to your LDAP server.

Note: A bind user must exist on the Windows server. This user is allowed to access and query the Windows server by using LDAP. If this user's password expires, change the password by editing `/etc/ldap.conf`.

To verify that you can query the Active Directory Server with the bind user, use the `ldapsearch` command. For example, assume that you use the user `ad` to query the Active Directory Server `16.1.2.3` in the domain `omgbl.atl.hp.com`. Run the following command:

```
ldapsearch -h 16.1.2.3 -x -W -D
"cn=ad,cn=Users,dc=omgbl,dc=atl,dc=hp,dc=com" -b
"dc=omgbl,dc=atl,dc=hp,dc=com" "objectclass=User" |more
```

The output prompts you for the bind user's password.

To configure PAM authentication by using LDAP, perform the following procedure:

1. Edit `ldap.conf` by adding the following lines:

   ```
   pam_login_attribute sAMAccountName
   binddn <bind user>
   bindpw <password for bind user>
   base <base DN>
   uri <ldap uri>
   ```

   For example:

   ```
   pam_login_attribute sAMAccountName
   binddn ad
   bindpw Password1
   base dc=omgbl,dc=atl,dc=hp,dc=com
   uri ldap://16.1.2.3/
   ```

2. Create `/etc/pam.d/ovo` with the following content:

   ```
   #%PAM-1.0
   auth        sufficient     pam_ldap.so
   auth        required       pam_deny.so
   account     required       pam_permit.so
   ```

3. Set up HPOM to use PAM authentication as described in "Configuring PAM User Authentication" on page 574.

Using Likewise Open with Windows Active Directory

Use this example to configure PAM authentication by using a third-party tool Likewise Open. Likewise Open is a free open source software that joins Active Directory domains. For more information, see the following Web site: http://www.likewise.com/products/likewise_open/

Note: The following procedure changes the authentication method of the whole system (for example, users will then be able to log on to the system by using `ssh` with their active directory credentials).

To configure PAM authentication by using Likewise Open, follow this procedure:

1. Download and install Likewise Open.
2. Join the systems that run HPOM to your Windows Active Directory as follows:

   ```
   /opt/likewise/bin/domainjoin-cli join <your domain> <domain user>
   ```
3. Create `/etc/pam.d/ovo` with the following content:

   ```
   #%PAM-1.0
   auth include system-auth
   account required pam_nologin.so
   account include system-auth
   password include system-auth
   session optional pam_keyinit.so force revoke
   session include system-auth
   session required pam_loginuid.so
   ```
4. Set up HPOM to use PAM authentication as described in "Configuring PAM User Authentication" on page 574.

Using Winbind with Windows Active Directory

Use this example to configure PAM authentication by using a free third-party tool Winbind. Winbind is a component of the Samba application suite that provides authentication of user credentials by using PAM. Winbind makes it possible to log on to a UNIX or a Linux system by using user and group accounts from an Active Directory domain. For more information, see the following Web site: http://technet.microsoft.com/en-us/magazine/2008.12.linux.aspx

Note: The following procedure changes the authentication method of the whole system (for example, users will then be able to log on to the system by using ssh with their active directory

credentials).

To configure PAM authentication by using Winbind, follow this procedure:

1. Install samba-common and samba-client with RID mapping support.
2. Start the RHEL graphical Authentication Configuration Tool (`system-config-authentication`) to configure Winbind.
3. Click the **User Information** tab, then click **Enable Winbind Support > Configure**, and then enter the following information:

   `Winbind domain:` **`your domain`**
   `Security model:` **`ads`**
   `Winbind ADS realm:` **`your.domain`** (for example, DNS domain)
   `Winbind Domain Controller:` **`*`** (or specify the domain controller)
   `Template shell:` **`/bin/bash`**

   Click **OK**.

   Caution: Do not click Join in this step.

4. Click the **Authentication** tab, and then click **Enable Winbind Support**.
5. Click **OK**. Now you can close system-config-authentication.
6. Configure home directories by editing `/etc/pam.d/system-auth`. Before the following existing line:

   ```
   session required pam_unix.so
   ```

   Add the following line:

   ```
   session optional pam_mkhomedir.so
   ```

7. Edit `/etc/samba/smb.conf` by adding the following line to the global section:

   ```
   template homedir = /home/%U
   ```

8. Edit `/etc/samba/smb.conf` by setting up ID mapping as follows:

   ```
   idmap backend = ad
   ```
   or
   ```
   idmap backend = rid
   ```

9. Join the domain as follows:

   Note: Before joining the domain, check time on your system and time on the domain controller. Time difference between the domain controller and your system must be less than five minutes.

   ```
   net ads join -U <domain user>
   ```

10. Create `/etc/pam.d/ovo` with the following content:

```
#%PAM-1.0
auth include system-auth
account required pam_nologin.so
account include system-auth
password include system-auth
session optional pam_keyinit.so force revoke
session include system-auth
session    required     pam_loginuid.so
```

11. Set up HPOM to use PAM authentication as described in "Configuring PAM User Authentication" on page 574.

Configuring PAM User Authentication by using LDAP on HP-UX with Windows Active Directory

To enable logon of the Windows users to the HP-UX systems that host HPOM, you can configure PAM authentication with Windows Active Directory. This section contains an example of using LDAP-UX as the LDAP client on HP-UX.

Using the LDAP Authentication with Windows Active Directory

Use this example to configure PAM authentication by using LDAP on HP-UX. This is required to enable access to Windows Server 2008 Active Directory from the HP-UX systems. Encrypted communication with the LDAP server is not covered with this example.

Make sure that the LDAP client software is installed on your system (LDAP-UX on HP-UX). The LDAP client should be configured to point to your LDAP server.

Note: A bind user must exist on the Windows server. This user is allowed to access and query the Windows server by using LDAP.

To verify that you can query the Active Directory Server with the bind user, use the `ldapsearch` command. For example, assume that you use the `ad` user to query the Active Directory Server 16.1.2.3 in the domain `omgbl.atl.hp.com`. Run the following command:

```
/opt/ldapux/bin/ldapsearch -h 16.1.2.3 -x -D
"cn=ad,cn=Users,dc=omgbl,dc=atl,dc=hp,dc=com" -w <password> -b
"dc=omgbl,dc=atl,dc=hp,dc=com" "objectclass=User" |grep dn: | more
```

To configure PAM authentication by using LDAP-UX on HP-UX, follow these steps:

1. Configure LDAP-UX by running the following commands:

   ```
   cd /opt/ldapux/config/
   ./setup
   ```

It is recommended to run the setup or the autosetup. A profile entry in the LDAP directory is created, which is in this case the Active Directory.

Make sure that you have an administrative AD account for the setup as well as an AD proxy account for the ongoing use. However, starting with version 5.0, the setup has an `-l` option for local only, which means that the profile is created locally and not in the Active Directory. If you use a local profile, you must start `ldapclientd` manually and make sure it is started at the system boot.

For details, see the LDAP-UX documentation.

2. Make sure the ADS users that need to log on to HPOM have the `uid` and `uidNumber` attributes.

   PAM uses the `uid` and `uidNumber` attributes (part of the `posixAccount` object class). When you create a user in the ADS, the `uid` and `uidNumber` attributes are not set by default. With Active Directory 2008, you can set them in the Active Directory Users and Computers application. To do this, follow these steps:

   a. Select the user, and then select **Properties**.

   b. Click the **Attribute Editor** tab. In this tab, you can modify the `uid` and `uidNumber` attributes.

      To check on the UNIX system that `uid` and `uidNumber` are set, you can use `ldapsearch`. For example:

      ```
      /opt/ldapux/bin/ldapsearch -h <ADS-server> -x -D
      "cn=ad,cn=Users,dc=omgbl,dc=atl,dc=hp,dc=com" -w <password> -b
      "dc=omgbl,dc=atl,dc=hp,dc=com" "cn=opc_op" | egrep "dn:|uid:|uidNumber:"
      ```

      An output similar to the following one appears:

      ```
      dn: CN=opc_op,CN=Users,DC=omgbl,DC=atl,DC=hp,DC=com
      uid: opc_op
      uidNumber: 777
      ```

3. Configure PAM on HP-UX. To do this, add the following lines to the `/etc/pam.conf` file:

   ```
   ovo auth required /usr/lib/security/hpux32/libpam_ldap.so.1
   ovo account required /usr/lib/security/hpux32/libpam_ldap.so.1
   ```

4. Enable PAM user authentication in HPOM by setting the `OPC_USE_PAM_AUTH` variable to `TRUE`:

   ```
   /opt/OV/bin/ovconfchg -ovrg server -ns opc -set OPC_USE_PAM_AUTH TRUE
   ```

   This setting instructs HPOM to use PAM as the authentication mechanism.

Configuring HPOM to Use an Active Directory Server as an LDAP Server on Solaris

To enable the logon of the Windows users to the Solaris systems that host HPOM, you can configure HPOM user authentication with Windows Active Directory. This section contains an example of using

LDAP.

Using LDAP Authentication with Windows Active Directory

Use this example to configure HPOM user authentication by using LDAP. This is required to enable access to Windows Server 2008 Active Directory from the Solaris systems. Encrypted communication with an LDAP server is not covered with this example.

Make sure that an LDAP client software is installed on your system. The LDAP client should be configured to point to your LDAP server.

To configure user authentication by using LDAP, follow these steps:

1. *Active Directory configuration*

   Make sure that the users that require access to HPOM are added to the Active Directory (AD), under the relevant AD group or groups.

   Note: It is not required that UNIX attributes are enabled for the users.

   To check if the `jdoe` user is added to the AD, use the `ldapsearch` command as follows:

   ```
   ldapsearch -b "OU=xyz,OU=abc,OU=AU,DC=domain,DC=subdomain,DC=company,DC=com"
   -x -h "10.10.10.10" -D "serviceaccount@domain" "sAMAccountName=jdoe"
   ```

2. *LDAP client configuration*

   Caution: Before running the `ldapclient` command, make a copy of the `nsswitch.conf` file because it is overwritten during the configuration of the LDAP client

   To configure the LDAP client, follow these steps:

   a. Initialize the LDAP client:

      ```
      ldapclient -v manual
                 -a credentialLevel=proxy
                 -a authenticationMethod=simple
                 -a "proxyDN=CN=serviceaccount,OU=ServiceAccounts,
                    DC=domain,DC=subdomain,DC=company,DC=com"
                 -a "proxyPassword=password"
                 -a "defaultSearchBase=OU=xyz,OU=abc,OU=AU,
                    DC=domain,DC=subdomain,DC=company,DC=com"
                 -a "defaultServerList=10.10.10.10"
                 -a domainName=domain.subdomain.company.com
      ```

   b. Configure the LDAP client with the required mappings:

      ```
      ldapclient mod -a
      servicesearchDescriptor=passwd:DC=domain,DC=subdomain,DC=company,DC=com?
      sub
      ldapclient mod -a
      ```

```
servicesearchDescriptor=group:DC=domain,DC=subdomain,DC=company,DC=com?s
ub
ldapclient mod -a attributeMap=passwd:uid=sAMAccountName
ldapclient mod -a attributeMap=passwd:uidnumber=uidNumber
ldapclient mod -a followReferrals=False
ldapclient mod -a attributeMap=passwd:gidnumber=gidNumber
ldapclient mod -a attributeMap=passwd:gecos=cn
ldapclient mod -a attributeMap=group:gidnumber=primaryGroupID
ldapclient mod -a attributeMap=shadow:uid=sAMAccountName
ldapclient mod -a
defaultServerList=server10.domain.subdomain.company.com
ldapclient mod -a objectClassMap=passwd:posixAccount=user
```

c. Use the `ldapclient list` command to verify the configuration.

d. Verify that `ldapclient` can connect to the AD and that it returns the information for the user.

Run the following command:

```
ldaplist -lv passwd jdoe
```

An output similar to the following one should appear:

```
+++ database=passwd
+++ filter=(&(objectclass=posixaccount)(uid=jdoe))
+++ template for merging SSD filter=(&(%s)(uid=jdoe))
dn: gecos=Doe,John,OU=,OU=,OU=,DC=domain,DC=subdomain,DC=company,DC=com
        objectClass: top
        objectClass: person
        objectClass: organizationalPerson
        objectClass: posixAccount
        cn: Doe, John
        sn: Doe
        givenName: John
        distinguishedName:
```

3. *nsswitch configuration*

The `nsswitch.conf` and `nsswitch.ldap` files do not require any manual modifications.

Caution: When you set up `ldapclient`, it overwrites the `nsswitch.conf` file. Therefore, make sure to restore the original file.

4. *PAM configuration*

Add the following lines to the end of the PAM configuration file:

```
ovo auth requisite pam_authtok_get.so.1
```

```
ovo auth required pam_ldap.so.1
ovo account required pam_ldap.so.1
```

5. *HPOM configuration*

   Configure HPOM to use LDAP for user authentication by running the following command:

   ```
   /opt/OV/bin/ovconfchg -ovrg server -ns opc -set OPC_USE_PAM_AUTH TRUE
   ```

   Caution: Run this command only on the active HP Operations management server.

   To test user authentication, you can log on to the Java GUI, and then check the `System.txt` file:

   ```
   tail -20 /var/opt/OV/log/System.txt
   ```

   An output similar to the following one should appear:

   ```
   0: INF: Wed Sep 18 19:16:34 2013: opcuiwww.ldap (27432/1):
   [OvCorCommRqstAuthGet.cpp:355]: Session 2110951664 started for user jdoe
   (parent pid=27421). (ITO217-11)
   0: INF: Wed Sep 18 19:16:51 2013: opcuiwww.sh(27421): opcuiwww exited
   normally. (ITO210-143)
   ```

   In this instance, `opcuiwww.ldap` indicates LDAP integration for user authentication.

Remote Access

This section describes security aspects for remote logon and command execution in UNIX environments. In this section, you can find the information about the application startup and the command broadcast, as well as the information about applications that enable user interaction in the Java GUI.

Application Startup and Command Broadcast

If HPOM operators do not log on with the default user account that is set up by the HPOM administrator, they must use the corresponding passwords for broadcasting commands or starting applications. If operators do not use the correct passwords, the command or application fails.

I/O Application Startup

When starting applications that require user interaction (for example, if they are configured as `Window (Input/Output)`), the operators must do one of the following:

- Specify passwords

  To do so, choose between silent and interactive application startup, for example:

- Silent startup:

  Specify any passwords the application requires in advance, for example, when configuring the application attributes.

- Interactive startup:

  Specify any passwords the application requires interactively, for example, whenever the application prompts the user (or script) for a password.

- Enable remote access

  To do so, add entries to the `.rhosts` file or modify the `/etc/hosts.equiv` file.

Password Assignment on Managed Nodes

This section explains how to assign passwords on UNIX and Microsoft Windows managed nodes.

Password Assignment on UNIX Managed Nodes

On UNIX managed nodes, the default HPOM operator `opc_op` cannot log on to the system using standard mechanisms such as `rlogin`, `telnet`, and so on because of a `*` entry in the `/etc/passwd` file and because `.rhosts` entries are not provided be default. If you want to provide a virtual terminal or an application startup that requires user input or output for the default HPOM operator, set a password or provide `.rhosts` or `/etc/hosts.equiv` functionality.

Note: The `opc_op` password should be consistent for all managed nodes.

For example, if *$HOME* is the home directory on the managed node, the *$HOME*`/.rhosts` entry of the executing user would look like the following:

<management_server> `opc_op`

In this example, *<management_server>* is the name of the machine hosting the HPOM management server. The name can be short of fully qualified depending on the configuration of your network.

Passwords Assignment on Windows Managed Nodes

On Microsoft Windows managed nodes, you can assign the password for the HPOM account during installation of the agent software. If you do not assign a password for the HPOM account, a default password is created. However, a password is not assigned by default.

Configuration Distribution

The command `opctmpldwn` provides a way of bypassing the standard mechanism for HPOM policy distribution and allowing you to download and encrypt HPOM policies and configuration data on the

management server and then copy the downloaded file to the target location on the managed nodes. Only assigned policies are downloaded (for example, log file, SNMP trap, `opcmsg`, threshold monitor, scheduled action, event correlation, and flexible management).

The downloaded files are encrypted either with the default key of the managed node or with keys generated specifically for the node.

For more information about command options and parameters, see the *opctmpldwn(1m)* manual page.

Automatic and Operator-Initiated Actions

Action requests and action responses can contain sensitive information (for example, application password, application responses, and so on) that might be of interest to intruders. In a secure system, this is not problem. However, if the requests and responses containing sensitive data have to pass through a firewall or over the Internet, where packets may be routed through many unknown gateways and networks, then you should take measures to improve security.

Shell Scripts

In addition, automatic actions and operator-initiated actions are normally executed as root. To prevent security holes, it is essential to protect any shell scripts (for example, those used to switch users) by assigning very restrictive permissions. You should also choose very carefully the commands that an application uses.

User Switch for HPOM HTTPS Agents

To further increase security, you can use the `ovswitchuser.sh` command to switch the user for HPOM HTTPS agents from user root to a specific user account or group.

For more information about command options and parameters, see the *ovswitchuser(1m)* manual page.

Remote Actions

HPOM offers a variety of security mechanisms that prevent the misuse of remote actions. The security measures are especially important for companies that use a single HP Operations management server to manage systems from more than one customer.

Remote actions designed for the managed nodes of one customer must not be allowed to be executed on the managed nodes belonging to another customer. Some of these security mechanisms are active by default. Other security measures must be enabled manually.

To prevent the interception and misuse of remote actions, HPOM offers the following security mechanisms:

- Assignment of configuration files:

  All HPOM configuration files on the managed nodes must belong to a trusted user. By default, this trusted user is the super user. You can change the trusted user (that is, the account under which the HPOM agents run) to another user. For more information about command options and parameters, see the *ovswitchuser(1m)* manual page.

- Encryption of message source policies:

  By default, HPOM encrypts all message source policies that are deployed on a managed node. Encryption protects message source policies from unwanted modifications and misuse.

- Prevention of remote actions:

  If necessary, you can entirely disable remote actions for all managed nodes.

  A remote action is defined as an automatic action or an operator-initiated action that is attached to an HPOM message sent by managed node A and configured to run on managed node B. The execution of such actions can be controlled by using the `remactconf.xml` file, which you can find in the following location:

  `/etc/opt/OV/share/conf/OpC/mgmt_sv/remactconf.xml`

- Detection of faked IP addresses or secret keys:

  If you installed the HPOM Advanced Network Security (ANS) extension, you can also check for mismatched sender addresses by running the following command on the HP Operations management server:

  ```
  # ovconfchg -ovrg <OV_resource_group> -ns opc -set OPC_CHK_SENDER_ADDR_MISMATCH
  TRUE
  ```

  In this instance, *<OV_resource_group>* is the name of the HP Operations management server resource group.

  If the check detects a mismatch between the IP address and the hostname (that is, any attempts to use faked IP addresses or secret keys that were generated by another node), all actions that are to be executed on the node are removed from the message. Only local actions that were already started on the message originator are not removed. Failed action requests are documented in annotations, which are added to the message automatically.

For more information about remote actions, see "Remote Action Authorization" on page 127 and the *HPOM Concepts Guide*.

Queue Files

The `opcmsg` and `opcmon` commands use the queue files for the message interceptor (`msgiq`) and the monitor agent (`monagtq`) to communicate with their corresponding processes. The queue files grant read or write permission to all users. You can read sensitive messages by displaying these queue files as a regular user.

Caution: The `opcmsg` and `opcmon` commands allow any user to send a message triggering an automatic action, even on another node.

Security in Flexible Management Environments

The use of certificate servers in the flexible management environments can be of the following types:

- "Environments Hosting Several Certificate Servers" on page 590
- "Shared Certificate Authority Scenario" on page 597

Environments Hosting Several Certificate Servers

It is possible that a managed environment has more than one certificate server. This situation is possible if two existing managed environments, both having an operating certificate server, are merged in a single environment.

Each certificate server uses a self-signed root certificate. As a result, all clients belonging to one certificate server do not trust any client belonging to the other. This can be solved by adding the root certificate of each certificate server to the trusted root certificates of the other certificate server. All agents in the managed environment are then triggered to receive the updated root certificate list from their certificate server.

If an agent is managed by multiple management servers the certificate management configuration is needed. By default, every HP Operations management server has its own Certificate Authority and the agent trusts only the certificates subscribed by this authority. For the flexible management environments, establish a trust between two or more managers so that their environments are able to communicate with each other.

The common scenarios are:

- "Merging Two Flexible Management Environments" on page 590
- "Certificate Handling for the Second HP Operations Management Server" on page 593
- "Shared Certificate Authority Scenario" on page 597

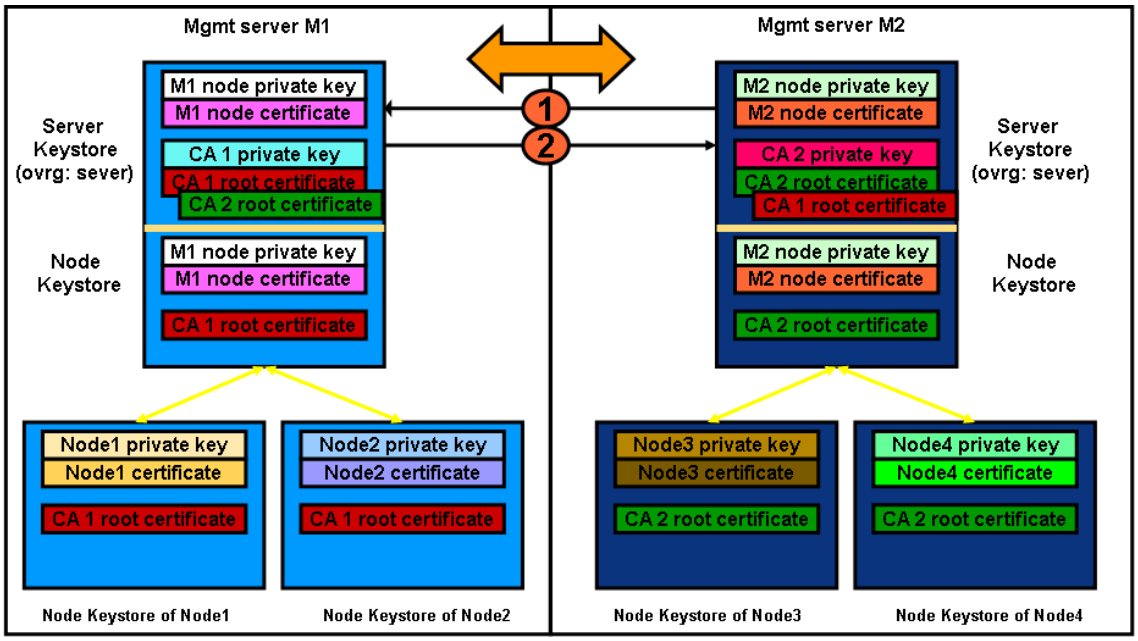
Merging Two Flexible Management Environments

Assume that there are two environments, one belonging to management server M1 with agents AM1 and the second belonging to management server M2 with agents AM2. Each management server has its own Certificate Authority.

To merge the environments, perform the following steps:

Note: HA environments and non-HA environments are handled in the same way. The following steps are valid for both types of installations.

1. Synchronize the trusted certificates on the management servers. M1 gets the root certificates of M2 and M2 gets the root certificate of M1.



   a. On management server M1, run the following command:

      `ovcert -exporttrusted -ovrg server -file <my_file>`

   b. Copy *<my_file>* to management server M2, by using, for example, `ftp`.

   c. On management server M2, run the following command:

      `ovcert -importtrusted -ovrg server -file <my_file>`

   d. On management server M2, repeat the same procedure.

   e. To verify that M1 and M2 have the root certificates of each other, run the following command on the both management server systems:

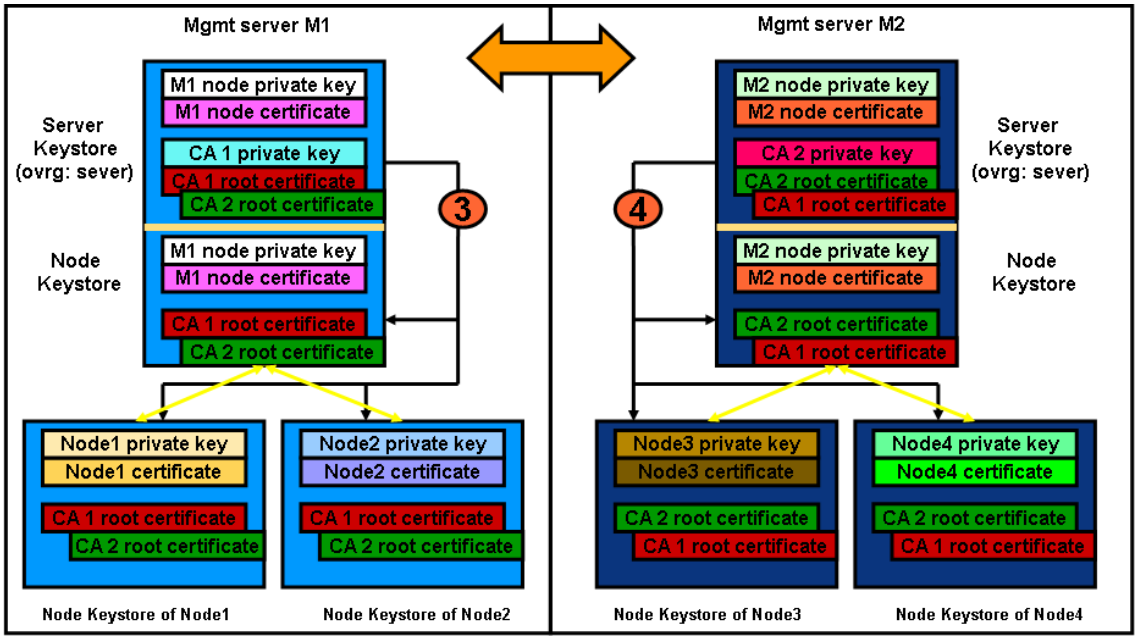
      `ovcert -list`

      Two trusted certificates should be listed.

2. Update the local root certificates on each managed node.

   To trigger this action on the managed node, run the following command:

   `ovcert -updatetrusted`



Note: In the cluster installations, the local certificates for agents and the management server are not the same.

On each management server M1 and M2, select all required managed nodes and run the application. The agents contact their certificate server for new root certificates.

To verify the root certificates on all managed nodes, run the following command:

```
ovcert -list
```

Two trust certificates should be listed.

3. Configure the management servers as regular nodes in the HPOM node banks of each other. M1 must be added to the node bank of M2 with its OvCoreId and M2 must be added to the node bank of M1 with its OvCoreId. To do this, use the following steps:

   a. Add node M1 to the node bank of M2, by running the following command on M1:

   ```
   opcnode -add_node node_name=<node_name_M2> net_type=<network_type> mach_type=<machine_type> group_name=<node_group_name>
   ```

   where *<node_name_M2>* is the name of management server M2.

   b. Add node M2 to the node bank of M1, by running the following command on M2:

   ```
   opcnode -add_node node_name=<node_name_M1> net_type=<network_type> mach_type=<machine_type> group_name=<node_group_name>
   ```

   where *<node_name_M1>* is the name of management server M1.

   c. Get and note down the OvCoreId of each management server (M1 and M2) by running the following command on M1 and M2:

   ```
   ovcoreid -ovrg server
   ```

d. Add the OvCoreId M1 to the database of M2 by running the following command on M2:

```
opcnode -chg_id node_name=<node_name_M1> id=<coreid_M1>
```

e. Add the OvCoreId of M2 into the database of M1 by running the following command on M1:

```
opcnode -chg_id node_name=<node_name_M2> id=<coreid_M2>
```

f. Verify that the nodes were correctly added to the databases by running the following command on M1 and M2:

```
opcnode -list_id node_list=<node_name>
```

On M1, the OvCoreId of node M2 should be listed.

On M2, the OvCoreId of node M1 should be listed.

Note: Do not forget to add uploaded nodes to the node group by using the `opcnode` command so that you are able to see messages. For more information, see the *opcnode(1m)* manual page.

4. Create or enhance the responsible manager policy on both servers and deploy it to their own agents.
5. Synchronize the node banks using `opccfgupld` and `opccfgdwn`. M1 gets the entries of M2, M2 gets the entries of M1 including their OvCoreIDs.

   By default, in the merged flexible environment all automatic and operator-initiated actions are allowed on both management servers because both management servers have root certificates installed and a trust relationship established. To restrict actions on management server from agents belonging to other management servers, set the following configuration setting:

   ```
   ovconfchg -ovrg server -ns opc -set OPC_RESTRICT_ACTIONS_WITH_FOREIGN_SIGNATURE TRUE
   ```

   If there are more than two servers in the flexible environment and you want to allow actions from the agents belonging to these servers, set the following configuration setting:

   ```
   ovconfchg -ovrg server -ns opc -set OPC_ACCEPT_ACTION_SIGNATURES_FROM <list_of_allowed_srv_COREIDs>
   ```

   where *<list_of_allowed_srv_COREIDs>* is a comma-separated list of other management servers CoreIDs.

   Note: This action restriction cannot be configured by using the `remactconf.xml` file because a trust relationship is established between servers through the installed root certificates.

Certificate Handling for the Second HP Operations Management Server

Assume that management server M2 has its own Certificate Authority and is used as a backup management server or a competence center. Assume that management server M1 owns agents AM1 and that management server M2 initially has no agents.

1. Synchronize the trusted certificates on the management servers by copying the root certificates of M2 to M1 and copying the root certificates of M1to M2. Complete the following steps:
   a. On management server M1, run the following command:

      ```
      ovcert -exporttrusted -ovrg server -file <my_file>
      ```

   b. Copy *<my_file>* to management server M2, for example, by using `ftp`.
   c. On management server M2, run the following command:

      ```
      ovcert -importtrusted -ovrg server -file <my_file>
      ```

   d. Repeat the same procedure for management server M2.
   e. To verify that M1 and M2 have the root certificate of each other, run the following command on both management servers:

      ```
      ovcert -list
      ```

      Two trusted certificates should be listed.
2. Update the root certificate on M1 by running the following command:

   ```
   ovcert -updatetrusted
   ```

   On M1, select AM1, and then run the application. The agent contacts its certificate server and asks for a new root certificate.
3. Configure the management servers as regular nodes in the HPOM node banks of each other. M1 must be added to the node bank of M2 with its OvCoreId and M2 must be added to the node bank of M1 with its OvCoreId. To do this, use the following steps:
   a. Add node M1 to the node bank of M2, by running the following command on M1:

      ```
      opcnode -add_node node_name=<node_name_M2> net_type=<network_type> mach_
      type=<machine_type> group_name=<node_group_name>
      ```

      where *<node_name_M2>* is the name of management server M2.
   b. Add node M2 to the node bank of M1, by running the following command on M2:

      ```
      opcnode -add_node node_name=<node_name_M1> net_type=<network_type> mach_
      type=<machine_type> group_name=<node_group_name>
      ```

      where *<node_name_M1>* is the name of management server M1.
   c. Get and note down the OvCoreId of each management server (M1 and M2) by running the following command on M1 and M2:

      ```
      ovcoreid -ovrg server
      ```

   d. Add the OvCoreId of M1 to the database of M2 by running the following command on M2:

      ```
      opcnode -chg_id node_name=<node_name_M1> id=<coreid_M1>
      ```

   e. Add the OvCoreId of M2 to the database of M1 by running the following command on M1:

      ```
      opcnode -chg_id node_name=<node_name_M2> id=<coreid_M2>
      ```

   f. Verify that the nodes were correctly added to the databases by running the following

command on M1 and M2:

```
opcnode -list_id node_list=<node_name>
```

On M1, the OvCoreId of node M2 should be listed.

On M2, the OvCoreId of node M1 should be listed.

Note: Do not forget to add uploaded nodes to the node group by using the `opcnode` command so that you are able to see messages. For more information, see the *opcnode(1m)* manual page.

4. Create or enhance the responsible manager policy on both management servers and deploy it to their own agents. M1 must deploy the responsible manager policy to all its managed nodes (in this case, they are M1 and AM1). M2 must deploy the responsible manager policy to its local agent if it was not already a part of the M1 environment.
5. Synchronize the node banks using `opccfgupld` and `opccfgdwn`. M2 receives all agents of M1, while M1 loads the local agent of M2, if it was not already present in the database.

Switching Certificate Authority Scenario

Assume that a flexible environment is already established. The sample flexible environment consists of the following:

- Management server A is an active Certificate Authority.
- Management server B is an alternative Certificate Authority.
- Systems are managed as HPOM nodes.
- Responsible manager policy was created and distributed to all managed nodes.

Switching Certificate Authority

To change the Certificate Authority in the sample flexible environment, perform the following steps:

1. Set management server B as a primary manager for a managed node by running the following command:

   ```
   opcragt -primmgr <node_name>
   ```
2. Remove all policies from this managed node that were distributed from management server A.
3. Stop the agent software on the managed node:

   ```
   ovc -stop
   ```
4. Remove the agent certificate from the managed node:

   ```
   ovcert -remove <alias>
   ```
5. Remove the trusted management server A certificate from the managed node:

   ```
   ovcert -remove <alias>
   ```

Note: Check that management server B is present on the node by using `ovcert -list`.

6. Issue a new certificate manually from management server B:

   ```
   opccsacm -issue -name <nodename> -file <filename> -coreid <OvCoreId>
   ```

7. Transfer the newly created certificate to the managed node.

8. Import the new certificate to the managed node:

   ```
   ovcert -importcert -file <filename>
   ```

9. Change the configuration settings on the managed node to reflect the change to the new Certificate Authority (management server B):

   ```
   ..
   [sec.cm.client]
   CERTIFICATE_SERVER=<management server B hostname>
   [sec.core.auth]
   MANAGER=<management server B hostname>
   MANAGER_ID=<management server B OvCoreId>
   ...
   ```

   The configuration settings are changed with the following commands:

   ```
   ovconfchg -ns sec.cm.client -set CERTIFICATE_SERVER <mgmt_srvB_hostname>
   ovconfchg -ns sec.core.auth -set MANAGER <mgmt_srvB_hostname>
   ovconfchg -ns sec.core.auth -set MANAGER_ID <mgmt_srvB_OvCoreId>
   ```

10. Start the agent software on the managed node:

    ```
    ovc -start
    ```

11. Distribute policies to the managed node.

12. *Optional.* Do not allow automatic or operator-initiated actions on the management server A.

    By default, in the merged flexible environment all automatic and operator-initiated actions are allowed on both management servers because both management servers have root certificates installed and a trust relationship established.

    To restrict actions on management server from agents belonging to other management servers, set the following configuration setting:

    ```
    ovconfchg -ovrg server -ns opc -set OPC_RESTRICT_ACTIONS_WITH_FOREIGN_SIGNATURE
    TRUE
    ```

13. *Optional.* Allow automatic and operator-initiated actions from the nodes belonging to the selected management servers on the management server A.

    If there are more than two servers in the flexible environment and you want to allow actions from the agents belonging to these servers, set the following configuration setting:

    ```
    ovconfchg -ovrg server -ns opc -set OPC_ACCEPT_ACTION_SIGNATURES_FROM <list_of_
    allowed_srv_COREIDs>
    ```

where *<list_of_allowed_srv_COREIDs>* is a comma-separated list of other management servers CoreIDs.

Note: This action restriction cannot be configured by using the `remactconf.xml` file because a trust relationship is established between servers through the installed root certificates.

Shared Certificate Authority Scenario

HPOM flexible management environment provides a possibility of working with only one Certificate Authority. However, this approach should be considered before setting up a flexible environment.

If you have an existing environment with two certificate authorities, it is not recommended to use the shared Certificate Authority scenario, as this would require you to replace all certificates that were granted by one of the Certificate Authorities. Consider that all management servers and their managed nodes are dependent on one Certificate Authority.

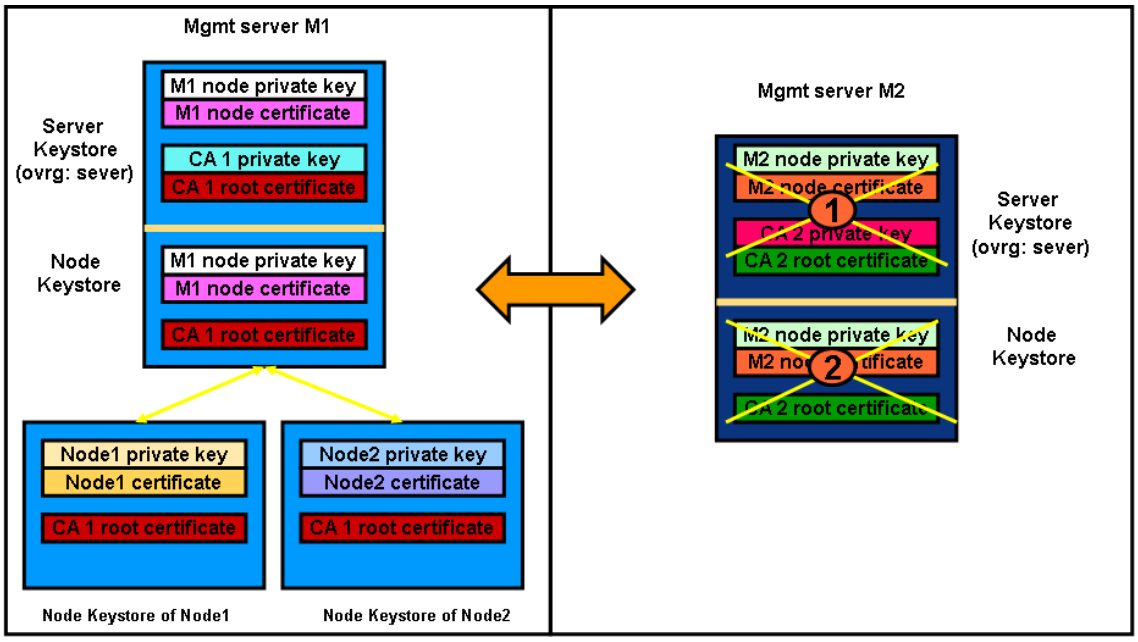
Establishing Shared Certificate Authority

Assume that the management server M1 has a Certificate Authority and M2 should not have one.

Perform the following steps:

1. Immediately after installing management server M2, remove the local certificates by running the following commands:

   ```
   ovcert -remove <cert_id>
   ovcert -remove -ovrg server <cert_id>
   ```



2. Add management server M2 to the node bank of M1.

   On node M1, run the following command:

   ```
   opcnode -add_node node_name=<node_name_M2> net_type=<network_type> mach_
   type=<machine_type> group_name=<node_group_name>
   ```

3. Create a certificate for M2 on M1 by running the following command:

   ```
   opccsacm -issue -name <node_name_M2> -coreid <core_ID_M2> -file <M2_cert> -pass
   <password>
   ```

   Note: To display the `OvCoreId` of M2, on the M2 system, run the following command:

   ```
   ovcoreid -ovrg server
   ```

   `opccsacm` also adds the OvCoreId of M2 to the database.

4. Copy the certificate to M2 and install it as the server certificate.

   - If M2 is an HP Operations HA cluster server, perform the following steps:

     i. Import the certificates, by running the following command:

     ```
     ovcert -importcert -ovrg server -file <my_cert> -pass <password>
     ```

     ii. Create an extra node certificate for each physical node, by running the following command on M1:

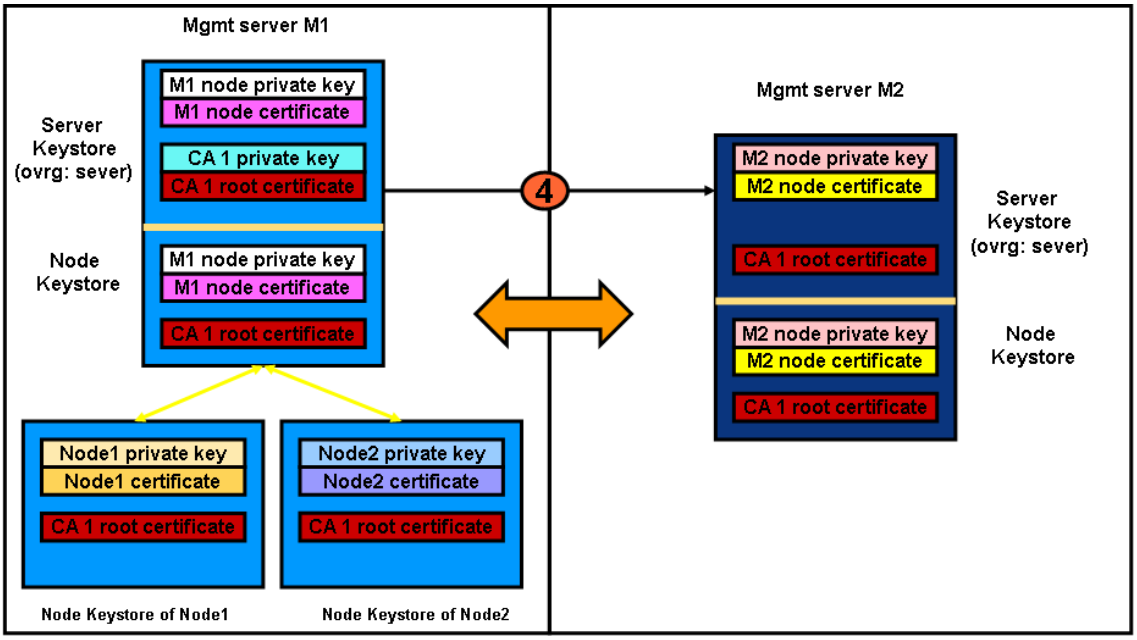
     ```
     opccsacm -issue -name <hostname_M2_cluster_node> -coreid <OvCoreId_M2_
     cluster_node> -file <my_cert> -pass <password>
     ```

     iii. Copy the node certificates to the M2 cluster nodes and install them by running the following command:

     ```
     ovcert -importcert -file <my_cert> -pass <password>
     ```

   - If M2 is not an HP Operations HA cluster server, run the following command:

```
ovcert -importcert -file <my_cert> -pass <password>
```



5. Instruct every managed node that is installed by M2 that its certificate server is M1 by placing an entry into the `bbc_inst_defaults` file. This file is used to automatically generate profiles for the agent installation. The location of the file is:

   `/etc/opt/OV/share/conf/OpC/mgmt_sv/bbc_inst_defaults`

   Note: If this file does not exist, create it by using the following sample file as a template:

   `/etc/opt/OV/share/conf/OpC/mgmt_sv/bbc_inst_defaults.sampl`

   Add the namespace and the certificate server specifications to your `bbc_inst_defaults` file as follows:

   ```
   [sec.cm.client]
   CERTIFICATE_SERVER <hostname_M1>
   ```

   For the local agent on M2, run the following command:

   ```
   ovconfchg -ns sec.cm.client -set CERTIFICATE_SERVER <hostname_M1>
   ```

6. On M1, specify the OvCoreId of M2 as a trusted OvCoreId, by running the following command:

   ```
   ovconfchg -ovrg server -ns opc -set OPC_TRUSTED_SERVER_COREIDS <M2 OvCoreId>
   ```

   If you have more than two management servers in your shared Certificate Authority flexible environment (for example, M1 with the Certificate Authority root certificate installed; M2 and M3 with a shared Certificate Authority issued by M1), complete the following steps:

   a. On M1, specify the OvCoreIds of M2 and M3 as trusted OvCoreIds:

      ```
      ovconfchg -ovrg server -ns opc -set OPC_TRUSTED_SERVER_COREIDS <OVCoreID_M2>,<OVCoreID_M3>
      ```

List all management server OvCoreIds in your flexible environment, excluding the OvCoreId from management server M1.

b. On all other management servers (for example, M2 and M3) with a shared Certificate Authority, specify a list of trusted OvCoreIds. This list should contain all management server OvCoreIds in this flexible environment, excluding the following:

- OvCoreId of the local management server where the following command is executed.
- OvCoreIds of management server M1 that has the Certificate Authority.

On M2, specify the OvCoreId of M3 as a trusted OvCoreId:

```
ovconfchg -ovrg server -ns opc -set OPC_TRUSTED_SERVER_COREIDS <OVCoreID_M3>
```

On M3, specify the OvCoreId of M2 as a trusted OvCoreId:

```
ovconfchg -ovrg server -ns opc -set OPC_TRUSTED_SERVER_COREIDS <OVCoreID_M2>
```

7. Unregister the Certificate Server (`ovcs`) component from M2 by running the following command:

```
ovcreg -del ovcs
```

8. Create or enhance the responsible manager policy on both servers and deploy it to their own agents. M1 must deploy a responsible manager policy to all of its agents that are to be managed by M2. M2 must deploy a responsible manager policy to its local agent, if it was not already a part of the M1 environment.
9. Download the node bank configuration on M1 and upload to M2 by using the `opccfgupld` and `opccfgdwn` commands.

HPOM Audits

HPOM 9.xx auditing is based on a series of entries to log files that are written when specific actions take place. These actions can be triggered either by internal processes, or by a user in one of the following ways:

- Java GUI:

  When users log on to HPOM with the Java GUI and use the Java GUI to perform operations.
- Command line utility (CLI):

  When users run administrator commands on the command line or call the commands from a script.
- Administration UI:

  When users log on to HPOM with the administrator's user interface and use the interface to perform operations.

Audit entries contain information indicating what kind of action took place, who performed it, when, and the audit area it concerns. Each entry has a default severity level, depending on the kind of action. The severity level can be `MINOR`, `MAJOR`, `SERIOUS`, or `INTERNAL` (the highest severity level).

Audit Levels

As an administrator, you can enable or disable the audit system. If the audit system is disabled, nothing is logged. When the audit system is enabled, you can choose the audit level (`OFF`, `MINIMAL`, `ADVANCED`, or `FULL`).

Note: The audit level `OFF` is not the same as disabling auditing. To disable auditing, use the `opcsrvconfig -audit -disable` command.

HPOM Audit System

To enable or disable the audit system in HPOM, use the `opcsrvconfig` command line utility as follows:

- To enable the audit system, run the following command:

  ```
  # opcsrvconfig -audit -enable <level>
  ```

- To disable the audit system, run the following command:

  ```
  # opcsrvconfig -audit -disable
  ```

For more information about the parameters and options you can use with the `opcsrvconfig` command, see the *opcsrvconfig(1m)* manual page.

The HP Operations management server checks the severity of the information written to the audit log file. Depending on the chosen audit level, entries with the specified severity are written to the `audit.opc.txt` file. For example:

`OFF`	Logs only entries with the severity `INTERNAL`
`MINIMAL`	Logs only entries with the severity `INTERNAL` and `SERIOUS`.
`ADVANCED`	Logs only entries with the severity `INTERNAL`, `SERIOUS`, and `MAJOR`.
`FULL`	Logs all entries.

Audit Entry Severity

Like the audit level, the severity level of a certain action can be customized. Each action has an XPL variable assigned. To set a custom value for this variable, run the following command:

```
# ovconfchg -ns audit -set <var> <sev_level>
```

In this instance, *<var>* is the name of the variable and *<sev_level>* is `MINOR`, `MAJOR`, `SERIOUS`, or `INTERNAL`.

For example:

```
# ovconfchg -ns audit -set OM_CFG_ADD_USER MAJOR
```

To list currently set variables, run the following command:

```
# opcsrvconfig -audit -list_custom
```

To obtain a list of all events and their default and current levels, run the following command:

```
# opcsrvconfig -audit -list_events
```

For more information about command options and parameters, see the *ovconfchg(1m)* and *opcsrvconfig(1m)* manual pages.

For more information about variables, see "HPOM Audits" on page 600.

Excluding Processes from Auditing

To exclude certain processes from auditing, use the `OMU_NO_AUDIT_PROCS` server configuration variable in the `audit` namespace. The `OMU_NO_AUDIT_PROCS` server configuration variable can contain a comma-separated list of processes that will be excluded from auditing.

For example, to exclude the `opccfguser` process from auditing, run the following command:

```
/opt/OV/bin/ovconfchg -ns audit -set OMU_NO_AUDIT_PROCS opccfguser
```

In this example, no audit appears for the `opccfguser` process, while other processes keep auditing correctly.

Setting a Size Limit for an Audit Log File

To set a size limit for an audit log file, `audit.opc.txt`, use the `OMU_AUDIT_LOG_MAXSIZE` server configuration variable in the `audit` namespace. This variable indicates the maximum log file size in kilobytes, up to a terabyte. When the `audit.opc.txt` file exceeds the specified size, it is moved to the `audit.opc.txt.xxx` file (`xxx` stands for `001` through `999`), where the contents are archived, and the `audit.opc.txt` file is started anew.

For example, to set the maximum log file size to four kilobytes, run the following command:

```
/opt/OV/bin/ovconfchg -ns audit -set OMU_AUDIT_LOG_MAXSIZE 4
```

Note: If the maximum log file size is not set correctly (for example, it is set to `-1`) or it is not set at all, or if the number of the old log files is 999, the contents of the `audit.opc.txt` file are not archived and the file itself grows indefinitely.

Audit Entry Format

All audit entries are written to the `/var/opt/OV/log/audit.opc.txt` file as the audit entry with the format illustrated in the following example.

Audit Log File Syntax

```
Time:<Time>|Sev:<Severity>|Area:<Area>|Action:<Action>|ID: (undefined)
|Source:OMU|OS User:<User>|App User:<User>|Text:<Text>[;<Param Type>:<Param
Value>
[;<Param 2 Type>:<Param 2 Value>...]]
```

The following list describes the parameters and variables used in this example:

<Time>	Time when the audit entry was logged.
<Severity>	HPOM severity level (`MINOR`, `MAJOR`, `SERIOUS` or `INTERNAL`).
<Area>	HPOM element or action on which the audit entry is based (Nodes, Policies, Server configuration, and so on).
<Action>	One of the following actions: `Read`, `Write`, `Execute`, `Start`, `Stop`, or `Login / Logout`.
<Source>	OMU
<OS User>	User who started the action logged in the audit log file.
<App User>	`opc_adm`, `opc_op`, or the HPOM user who created the action. If not known, `N/A` or `Admin (N/A)` is shown.
<Text>	Text describing the action that caused that entry in the audit log file.

Audit Areas

The following list provides a complete overview of all the areas covered in an HPOM audit:

- Functional:
  - Audit:
    - Startup
    - Shutdown
    - Config
  - Authorization:
    - Logon
    - Logout
    - HPOM user

        - HPOM user profile
        - HPOM certificate actions
    - HPOM objects:
        - HPOM message
        - HPOM node
        - HPOM application
        - External application
        - HPOM configuration
        - Other HPOM objects
    - HPOM database:
        - Read
        - Write
- HPOM database:
    - Read
    - Write
- HPOM file access:
    - HPOM script/binary access:
        - Read
        - Write
        - Execute
    - HPOM configuration file access:
        - Read
        - Write
        - Execute
- HPOM processes:
    - Startup
    - Shutdown

Start-up Messages

According to the National Institute of Standards and Technology (NIST) 800-37 standard, usage and criticality of any application should be acknowledged before its startup, as well as allowance for its usage. Acknowledgement is achieved with a warning message that is displayed before the application starts.

A start-up message does not exist by default in the Java GUI or the Administration UI. You can create it by writing your own text in a text editor and storing the message in the database. You can also set and change its status (that is, enable it or disable it). For details, see "Creating a Start-up Message" on page 605.

If the start-up message is enabled, it appears after the Log-on window. If the agreement defined in this message is accepted, HPOM starts. Otherwise, the log-on sequence stops immediately.

If the start-up message is disabled, HPOM starts right after the Log-on window.

Note: You can resize the Startup Message window according to your preferences.

Creating a Start-up Message

Before you create a start-up message, consider the following points:

- Customization:

  The start-up message is defined and enabled after the HPOM installation.

  You must be the `root` user to customize, edit, or change the status of the start-up message.
- Database storage:

  The start-up message is stored in the `opc_mgmtsv_config` table in the `ovou_license_text` attribute.

  For details about the database tables, see the *HPOM Reporting and Database Schema*.

Creating a Start-up Message

To create a start-up message, follow these steps:

a. Write your own message in a text editor and save it in a file.

   The length of the message may not exceed 2048 single-byte characters or 1024 multi-byte characters.

   To ensure that the start-up message is displayed correctly in the start-up message window, make sure you consider the line fields in the text editor while composing the message.

b. Enable the new custom message.

To read the customized start-up message from a file, store the message in the HPOM database, and enable it for use in the Java GUI or Administration UI by using the `opcuistartupmsg` command line tool:

```
/opt/OV/bin/OpC/opcuistartupmsg -f <filename> -e
```

For more information about the `opcuistartupmsg` tool, see the *opcuistartupmsg(1m)* manual page.

c. *Administration UI only:* Complete enabling the new custom message by following these steps:

   i. Set the `STARTUP_MESSAGE_VISIBLE` server configuration variable to `TRUE` by running the following command:

   ```
   ovconfchg -ovrg server -ns adminui -set STARTUP_MESSAGE_VISIBLE TRUE
   ```

   ii. Restart all HPOM processes by running the following command:

   ```
   /opt/OV/bin/ovc -restart
   ```

To display the current start-up message and its status, use the `opcuistartupmsg -s` command.

HPOM FIPS Compliance

Federal Information Processing Standard (FIPS) is a series of security standards, which establish requirements for ensuring computer security and interoperability. HPOM can be configured to make it compliant with FIPS 140-2. FIPS 140-2 focuses on the use of cryptography modules defined by the National Institute of Standards and Technology (NIST), to protect unclassified, valuable, and sensitive information in hardware and software products. A cryptographic module must have a validation certificate provided by the Cryptographic Module Validation Program (CMVP).

Caution: You cannot revert FIPS mode. After you configure HPOM in FIPS mode, you cannot reconfigure HPOM to standard non-FIPS mode. To run HPOM in non-FIPS mode, you must reinstall and configure HPOM. For information about installing and configuring HPOM, see the *HPOM Installation Guide for the Management Server*.

Considerations Before Running HPOM in FIPS Mode

Note the following considerations before configuring HPOM to run in FIPS mode:

- You cannot revert FIPS mode. You must reinstall HPOM to switch to non-FIPS mode.
- FIPS does not support encryption with a key length of less than 2048 bits.
- FIPS mode does not enforce encryption. However, when encryption is used, only approved algorithms are allowed. HPOM automatically uses FIPS-compliant cryptographic methods when HTTPS communication is enabled.
- You must manually import the certificate for TLS communication with the database.

- Integrations:

  When you enable FIPS mode for HPOM, all the existing data is decrypted and encrypted using a FIPS-compliant cipher. New data is directly encrypted using a FIPS-complaint cipher. For the new encrypted data, some fields of the database require modification to accept large data. This modification is done automatically during FIPS configuration.

Prerequisites to Configure HPOM for FIPS Compliance

- Make sure that encryption, decryption, and hashing use only validated cipher methods, which consist of symmetric keys, asymmetric keys, and secure hash standard. For more information, see *Annex A: Approved Security Functions* for FIPS 140-2 at http://csrc.nist.gov/publications/PubsFIPS.html.

Requirements to Configure HPOM for FIPS Compliance

When configuring HPOM for FIPS compliance, make sure that you meet the following requirements.

Caution: For the entire HPOM managed environment to be FIPS-compliant, all the components such as HP Operations Agent, HP Performance Manager, and so on in the managed environment must also be FIPS-compliant.

Server Requirements

Make sure that the systems on which you plan to run HPOM in FIPS mode meet the hardware and software requirements listed in the *HPOM Installation Guide for the Management Server*. In addition to these requirements, you must install HPOM 9.22 or upgrade HPOM to version 9.22 on these systems.

Agent Node Requirements

Install HP Operations Agent version **12.01** or **higher** on the managed nodes and on the management server as a managed node. Before configuring the HPOM server, make sure that all managed nodes are FIPS enabled.

Database Requirements

When configuring HPOM for FIPS compliance, you must use Oracle 12c. To make the communication between HPOM and DB FIPS-complaint, you must use Oracle 12c.

Note: For FIPS-compliant secure connection to an Oracle database in an HPOM managed environment, you must configure the Oracle database to use TLS/SSL. You must also configure HPOM and the Administration UI to connect to the database by using TLS/SSL. For more

information, see the *HPOM Installation Guide for the Management Server* that is available at: https://softwaresupport.hpe.com/group/softwaresupport/manuals.

Certificate Requirements

The server or client certificates used for communication must be FIPS-compliant. The RSA keys for asymmetric encryption must have a minimum length of 2048 bits.

Client Requirements

Install JRE 1.80_92 or higher on Java GUI client systems.

Limitations of HPOM in FIPS Mode

The following limitations apply when HPOM runs in FIPS mode:

- The Administration UI, Java GUI, and Web Services take a longer time to be launched when HPOM is operating in FIPS mode, as compared to the non-FIPS mode.
- In a FIPS-compliant HPOM environment, you cannot launch the Java GUI by using the following command:

  ```
  /opt/OV/bin/OpC/ito_op -https true
  ```

  You must launch the Java GUI by using WebStart or install the Java GUI client.

Configuring HPOM for FIPS Compliance

Use the following command to enable FIPS:

```
/opt/OV/bin/OpC/opcfips enable
```

The above command performs the following steps to enable FIPS mode on the management server.

1. Enables FIPS mode on the management server by configuring HPOM and local agent to use FIPS compatible algorithms for encryption.
2. Converts the Tomcat keystore that contains the certificate which is used to encrypt HTTPS communications to `PKCS12` format.

   Note: This step may take a few minutes to complete.

3. Enables FIPS mode in the Administration UI by configuring the Administration UI certificate stores to be FIPS-compliant.

   Note:

`opcfips` tool converts only default certificates and Administration UI passwords to FIPS-complaint form.

For non-default Administration UI configuration passwords, update the passwords to FIPS-complaint format by running the following command:

```
/opt/OV/OMU/adminUI/adminui password -u
```

If the Administration UI uses a third-party certificate, convert the certificate to FIPS-complaint format by running the following command:

```
/opt/OV/nonOV/jre/b/bin/keytool -importkeystore -srckeystore source_
keystore.jks -srcstoretype JKS -deststoretype PKCS12 -destkeystore
destination_keystore.p12 -srcstorepass password -deststorepass password
```

After the `opcfips` tool completes these steps, the HPOM process are restarted in FIPS mode as root user.

When you configure a management server that is operating in non-root mode to be FIPS-compliant, the HPOM processes restart in the root mode after FIPS is configured. To restart the HPOM processes in non-root mode, perform the following steps:

1. As a root user, stop all HPOM processes by running the following command:

   ```
   ovc -kill
   ```
2. Log on as a non-root user:

   ```
   su - opc_op
   ```
3. Start all HPOM processes by running the following command:

   ```
   ovc -start
   ```

Configuring HPOM for FIPS Compliance in a Cluster Environment

Perform the following steps on HPOM.

1. On the Active node, disable the switchover from HPOM by running the following command:

   ```
   /opt/OV/lbin/ovharg -monitor ov-server disable
   ```
2. Enable FIPS on the Active node by running the following command:

   ```
   opcfips enable
   ```
3. On the Active node, enable the switchover from HPOM by running the following command:

   ```
   /opt/OV/lbin/ovharg -monitor ov-server enable
   ```
4. Switch to the Passive node and enable FIPS by running the following commands:

   ```
   /opt/OV/bin/ovharg_config ov-server -switch <Passive Node>
   perl /opt/OV/lbin/secco/FIPS_tool -enable_FIPS
   ```

Manually Converting HPOM Data for FIPS Compliance

In instances such as restoring an HPOM database from a time before FIPS was enabled, passwords are encrypted in the old format. The HPOM managed environment is hence not FIPS-compliant.

You can manually convert the old data for FIPS compliance, by running the following command:

```
/opt/OV/bin/OpC/opcfips convert enable
```

Downloading Configuration for Older HPOM Versions

When HPOM runs in FIPS mode, it writes all encrypted passwords to the configuration download files in a FIPS-compatible format. These configuration files can be directly uploaded to any server with HPOM version 9.22 or higher installed, even if the server is not FIPS-compliant. However, older HPOM versions are not capable of decoding FIPS-compliant passwords. To export FIPS-compliant data for an HPOM version older than 9.22, perform the following steps:

1. Disable conversion of existing data to encrypted form by running the following command:

   ```
   /opt/OV/bin/OpC/opcfips convert disable
   ```
2. Perform configuration download.
3. Enable conversion of existing data to FIPS-compatible format by running the following command:

   ```
   /opt/OV/bin/OpC/opcfips convert enable
   ```

Chapter 15: Smart Card Authentication

In This Chapter

This chapter describes how to configure HPOM to provide smart card authentication. A smart card is a physical device containing one or more user certificates that are used for identification in secure systems.

In this chapter, you can also read how secure communication is established and which events constitute an HPOM smart card session.

For detailed descriptions, see the following sections:

Smart Card Authentication on HPOM

HPOM supports certificate technologies to authenticate and authorize users. By configuring smart card authentication on HPOM, access to HPOM user interfaces is allowed only to operators who possess a valid certificate. Therefore, security is increased and access procedures are simplified.

When configuring HPOM to use smart card authentication, you must first set up an environment, and then customize access rights for each user. For details, see "Configuring Smart Card Authentication on HPOM" on page 612.

Authentication and Secure Communication

Authentication plays a vital role in ensuring that system access is as secure as possible. In response to increased security requirements, HPOM can be configured to use certificates instead of the standard model where each user enters a user name and a password manually. This means that two-factor authentication (that is, requiring a PIN entry as well as possession of a valid certificate contained on the card) can now be used instead of one-factor authentication (that is, using something known only to the user).

With smart card authentication, the certificate stored on the card is checked for a valid expiration date, and then against the certificate authority server to verify that it is not revoked.

The main communication security components responsible for creating and managing certificates are a certificate server, a keystore, and a certificate client. The following conditions are required for secure communication:

- The server system hosts the certificate server that contains the needed certification authority (CA) functionality.
- Each system that is involved in communication has a certificate that was signed by the certificate server with the CA private key.
- Each system has a list of trusted root certificates that must contain at least one certificate. The trusted root certificates are used to verify the identity of the communication partners. A communication partner is trusted only if the presented certificate can be validated using the list of trusted certificates.

Secure Data Exchange

To provide secure data exchange, HPOM uses asymmetric encryption. This means that two related keys—a key pair—are used to encrypt a message. The key pair consists of a public key and a private key. The intended recipient's public key is available to anyone who wants to send a message, whereas the private key that is needed for decryption of the message is known only to the receiver. Therefore, the message that is encrypted by using the public key can only be decrypted by using the corresponding private key and vice versa.

Configuring Smart Card Authentication on HPOM

To configure smart card authentication on HPOM, complete these tasks:

- Task 1: "Setting up the HP Operations Management Server for Smart Card Authentication" on page 612
- Task 2 *(optional)*: "Customizing Access Rights" on page 616

Setting up the HP Operations Management Server for Smart Card Authentication

For HPOM to support smart card authentication, the HP Operations management server must be set up. The initial setup of the environment is done by using the `scsetup` script, the syntax of which is as follows:

```
scsetup enable [-c <certificate_directory>]
        disable
        authgrp [<group>]
```

Note: If you want to specify a certificate directory containing root certificates, use the `-c`

`<certificate_directory>` option with `scsetup enable`. Otherwise, a default set of root certificates is used.

You can choose among the following log-on modes:

- `enabled`

  Enables a logon to the HPOM user interfaces by using only a smart card for authentication.
- `legacy`

  Enables a logon to the HPOM user interfaces either by providing a user name and a password or by using a smart card for authentication.
- `disabled`

  Enables a logon to the HPOM user interfaces by providing only a user name and a password.

To set up the HP Operations management server for smart card authentication, follow these steps:

1. Enable smart card authentication by running the `scsetup enable` command.

   For detailed information about the actions that are performed during the Java GUI or Administration UI setup, see "Java GUI and Administration UI Setup" on page 614.
2. *Optional:* Enable command line authentication.

   The `scsetup authgrp <group>` command allows you to specify an operating system user group authorized to run some of the command line interfaces that required the `root` user before the smart card feature was supported with HPOM.

   Note: If you run `scsetup authgrp` only (that is, without specifying the desired authorized group), this option is disabled.
3. *Optional:* Customize the smart card authentication options.

   For detailed information, see "Customizing Access Rights" on page 616.
4. Start the HP Operations management server processes by running the following command:

   `/opt/OV/bin/ovc -start`

Configuration Files

Before you start configuring smart card authentication on HPOM (that is, before you run the `scsetup` script), you can find all configuration files at the following location:

`/var/opt/OV/conf/webserver/`

Note: Smart card authentication is disabled by default. After you run the `scsetup` script, the configuration files are moved to the correct location (for example, the web server configuration directory, HTML document directories, and so on).

The following is the list of all configuration directories:

`certificates`	Contains all root certificates that are trusted. All certificates used for authentication must be directly or indirectly signed by these certificates.
`java-tomcat7`	Contains the filters that are attached to the Tomcat web server so that the certificates are accepted and their owner is mapped to a particular HPOM user. Some of these filters also include the source code so that you can provide your own user mapping implementation.
`perl`	Contains an alternative implementation of the Java GUI's Perl launcher that allows smart card authentication.

Java GUI and Administration UI Setup

The Java GUI and Administration UI setup actions are performed in the following order:

1. *Enabling certificate revocation checks in the HPOM Tomcat or Jetty web server*

   Even if the certificate is valid and not expired, it could be revoked because of one or more of the following reasons:

   - The CA issued a wrong certificate.
   - The owner's contract is terminated.
   - The owner misused the certificate or failed to adhere to CA policies.
   - The private key is leaked or otherwise compromised.

   In any of these cases, the certificate cannot be trusted anymore. Because of this, HPOM must check the certificate against a list of revoked certificates. These lists are published by the CA and renewed regularly.

2. *Creating a certificate keystore*

   A keystore is a file that serves as a container for one or more certificates. The keystore created by using the `scsetup` script contains the certificates for all certification authorities. Therefore, the Tomcat or Jetty web server can check if the certificate provided by the user is valid.

3. *Copying the default mapping library to the Tomcat or Jetty library directory*

   When the certificate is accepted, a method for determining if a user has access to the application must be established. In addition, the certificate user must be mapped to the HPOM user.

   When a user logs on to the Java GUI or the Administration UI with a certificate, a mapping procedure that assigns a certificate user to a particular Java GUI or Administration UI user is required. By default, the mapping procedure is done in the `com.hp.ov.tomcat.UserAuth` class in the `OmRealm.jar` file. If the `com.hp.ov.tomcat.UserAuth` class is used, the

`/etc/opt/OV/share/conf/OpC/mgmt_sv/SC_JGUI_users` file (for the Java GUI) or the `/etc/opt/OV/share/conf/OpC/mgmt_sv/SC_ADMINUI_users` file (for the Administration UI) must be created. This file contains a series of text lines that indicates a certificate's common name and assigns the Java GUI or Administration UI user to it. For example:

```
John Doe=opc_adm
John Smith=john_smith@mycompany.com
```

Therefore, you can add or reassign users by manually editing this file.

If you want to implement a different or more complex authentication system, you must create a custom class named `com.auth.custom.UserAuth` in a jar file, and then copy that jar file to the `/opt/OV/nonOV/tomcat/b/libs` directory (for the Java GUI) or the `/opt/OV/OMU/adminUI/lib/midas` directory (for the Administration UI).

Caution: *Administration UI only*: Make sure that the name of the jar file is `user-auth.jar`

The class must contain a single, static method that is in accordance with the following definition:

```
public static String authorizeUser (X509Certificate userCert){}
```

This method processes the provided user certificate and returns the corresponding Java GUI or Administration UI user name. If the provided user certificate is incorrect or its user is not authorized, the method returns null.

After deploying the custom class, make sure to reload the Java GUI or Administration UI configuration by running the following commands:

- For the Java GUI:

  ```
  /opt/OV/bin/ovc -stop ovtomcatB
  /opt/OV/bin/ovc -start ovtomcatB
  ```

- For the Administration UI:

  ```
  /opt/OV/OMU/adminui clean
  /opt/OV/OMU/adminui start
  ```

4. *Replacing authentication procedures in the Java GUI or Administration UI web applications*

   The Java GUI or Administration UI web application is modified to enable certificate authentication. In addition, a smart card filter is enabled. This filter checks if the provided certificate is stored in a smart card and if it is intended for authentication purposes.

5. *Java GUI only: Replacing the Java GUI Perl launcher with a launcher with smart card suppor*t

   The Java GUI is run as an applet by using the `ito_op_applet_cgi.ovpl` Perl script. A modified script, `ito_op_applet_cgi_sc.ovpl`, which receives the certificate information for Tomcat, encrypts the information, and sends it to the Java GUI, is provided. The Java GUI receives this authentication information and an automatic logon is performed (that is, with no need to provide a user name and a password).

6. *Setting up required HP Operations management server configuration variables*

   The `scsetup` script creates several variables that are read by HPOM processes. These variables influence the behavior of the processes. For example, some of these variables represent the mode of operation and a timeout that indicates the maximum period of time that is allowed to pass from the moment the user enters a certificate until the connection attempt takes place. If the timeout is too long, the certificate is rejected and the user must provide it again.

   For detailed information, see "Customizing Access Rights" on page 616.

7. *Java GUI only: Configuring the Tomcat HTTPS port to enable certificate authentication*

   The Tomcat HTTPS port must also be modified so that it can access the keystore and ask the user for a matching certificate.

8. *Restarting the Tomcat or Jetty web server*

   A restart of the `ovtomcatB` or `adminui` service is required for HPOM to accept the new configuration.

Customizing Access Rights

When customizing access rights, you can set the following variables:

`Mode`	Specifies if smart card authentication is enabled. The possible values are as follows: • `enabled` • `legacy` *(the default value)* • `disabled` For details, see "Setting up the HP Operations Management Server for Smart Card Authentication" on page 612. If you want to change the mode manually, you can only change it from `enabled` to `legacy` and vice versa. To do so, run the following command: `/opt/OV/bin/ovconfchg -ovrg server -ns SC -set Mode <value>` In this instance, `<value>` is either `enabled` or `legacy`. Changing the mode from `enabled` or `legacy` to `disabled` and vice versa can only be done by using the `scsetup` script.
`TokenTimeout`	Indicates the maximum period of time that is allowed to pass from the moment the user enters a certificate until the connection attempt takes place (that is, the period during which the certificate is still valid). The default value is 15 seconds and the time format is set to 00h00m00s (for example, 15h12m13s).

OPC_JGUI_TIMEOUT	Represents the number of minutes after which the user that is inactive for an extended period of time is logged out of the Java GUI. To set the desired number of minutes, run the following command: `ovconfchg -ovrg server -ns opc -set OPC_JGUI_TIMEOUT <number_of_minutes>` For example, for a user to be logged out of the system after being inactive for one minute, run the following command: `ovconfchg -ovrg server -ns opc -set OPC_JGUI_TIMEOUT 1`
AuthGrp	Checks if a user belongs to a specified operating system group (the default value is `SCauth`). If the user belongs to the specified operating system group that is stored in the `AuthGrp` variable, access is granted. Changing this variable can only be done by using the `scsetup` script.
EnableFilters	If enabled, additional checks are done for the provided certificate to make sure that it is present on a smart card and is meant to be used only for authentication.

Structure of an HPOM Smart Card Session

The following represents the sequence of events for a user's HPOM smart card session:

- Each smart card session begins when the user inserts the smart card into the card reader and validates it by entering the correct PIN.
- After the user enters the link to access the HPOM user interface, identification is requested.

  In the User Identification Request window, the user must choose the correct certificate, and then click OK.

  The certificate is validated by the following steps:

  - The certificate is read from the smart card.
  - The certificate is verified. It must be created by a trusted CA, and it may not be expired or revoked.
  - The certificate user is mapped to the HPOM user.
- The authorization information is sent to the server that allows the HPOM user interface to establish a connection. When the connection is established, the user is logged on to the HPOM user interface.
- The smart card session ends when the user logs out of the Java GUI or the Administration UI, or closes the browser window.

Viewing Log Files

When troubleshooting, you can use a log file analysis that represents a useful methodology for understanding all the aspects of smart card authentication on HPOM as well as for investigating the cause of problems. Log files can help you pinpoint when and where problems occurred.

Depending on what you want to view, choose one of the following log files:

- `/var/opt/OV/log/tomcat/ovtomcatb.out`

  Used for viewing the debug of Tomcat web server's authentication (that is, provided certificates, validation, results, possible errors, and so on).
- `/opt/OV/OMU/adminUI/logs/web.log`
  `<Jetty_home>/logs/jetty*.log`

  Used for viewing the debug of Jetty web server's authentication (that is, provided certificates, validation, results, possible errors, and so on).

Part VI: Localization Support

HP Operations Manager (HPOM) provides local language support for the HPOM server processes, managed node commands and processes, and Java GUI.

For more information, see "HPOM Language Support" on page 620.

Chapter 16: HPOM Language Support

In this Chapter

The information in this chapter describes the language dependencies of the HP Operations Manager (HPOM) management server processes, managed node commands and processes, and the Java GUI. You can also find information about the languages, `LANG` settings, and character sets that the various HPOM platforms support. In this chapter, you can find information covering the following topics:

Language Support on the Management Server

On the HP Operations management server, localization considerations determine the following:

- Language:

  Language used to display status messages from the HP Operations server and managed nodes in the HPOM Java GUI.

- Character set:

  Character set used for internal processing.

Setting the Language on the Management Server

HPOM uses the `LANG` environment variable to determine the language of the message catalog and most HP Operations management server processes.

When you start the HP Operations management server processes (with `ovc -start` or `opcsv -start`), HPOM evaluates the currently set locale and selects the related message catalog to be used. The evaluation and the selection usually take place during the system boot.

The `opcsv -start` command is run on the management server from within the shell script `omu500`, which you can find in the following location:

- HP-UX:

  `/sbin/init.d/omu500`
- Linux:

  `/etc/rc.d/init.d/omu500`
- Solaris:

  `/etc/init.d/omu500`

The locale of the management server is specified by the configuration setting `ctrl.env:LANG` and set automatically during installation and setup of the HPOM management server. You can view the current setting using the `ovconfget` command, as follows:

```
# /opt/OV/bin/ovconfget ctrl.env LANG
```

Environment variables specified in the `ctrl.env` name space overwrite the values set by the system boot scripts and, in addition, (respectively also from the user's context if started manually) for all HPOM processes, that is: everything controlled by the `ovc` daemon, `ovcd`.

The configuration setting `ctrl.env:LANG` should always point to a Unicode locale such as `en_US.utf8` or `ja_JP.UTF-8`. (character-set values depend on the operating system) to ensure that the HPOM server runs in Unicode. Since ASCII is a subset of UTF8, this is also true for English environments.

Note: If the configuration setting `ctrl.env:LANG` does *not* point to a Unicode locale, multi-byte characters are ignored on the management server, for example, when a Japanese agent sends messages to an HPOM management server using LANG=C, or if LANG is not set at all.

If necessary, you can change the language setting on the HPOM management server using the `ovconfchg` command, as follows:

```
# /opt/OV/bin/ovconfchg -ns ctrl.env -set LANG <desired_locale>
```

Types of Language Variables for the Management Server

HPOM supports only UTF-8 encoding. UTF-8 encoding enables the usage of multilingual characters in different HPOM elements, and eliminates the problems associated with character set incompatibility. You must set up a UTF-8 based locale if you want to ensure that the HPOM management server functions correctly.

The settings for the `LANG` variable listed in Table 46 are supported for the management server. HPOM has been verified to run in these languages.

Table 46: LANG Settings for the HP Operations Management Server

Language	LANG		
	HP-UX	**Linux**	**Solaris**
Czech	cs_CZ.utf8	cs_CZ.UTF-8	cs_CZ.UTF-8
English	C [a] C.utf8 en_US.utf8	C a en_US.UTF-8	C a en_US.UTF-8
French	fr_FR.utf8	fr_FR.UTF-8	fr_FR.UTF-8
German	de_DE.utf8	de_DE.UTF-8	de_DE.UTF-8
Italian	it_IT.utf8	it_IT.UTF-8	it_IT.UTF-8
Spanish	es_ES.utf8	es_ES.UTF-8	es_ES.UTF-8
Japanese	ja_JP.utf8	ja_JP.UTF-8	ja_JP.UTF-8
Korean	ko_KR.utf8	ko_KR.UTF-8	ko_KR.UTF-8
Russian	ru_RU.utf8	ru_RU.UTF-8	ru_RU.UTF-8
Simplified Chinese	zh_CN.utf8	zh_CN.UTF-8	zh_CN.UTF-8
Traditional Chinese	zh_TW.utf8	zh_TW.UTF-8	zh_TW.UTF-8

Setting the Database Character Set on the Management Server

The database character set specified during the HPOM installation determines the character set used for internal processing of data on the management server. Note that HPOM supports only UTF-8 encoding for the database (`AL32UTF8` for the Oracle database and `UTF-8` for the PostgreSQL database). Therefore, all data on the management server must be entered with UTF-8 encoding.

In most cases you can accept the default value for the Oracle database character set used by HPOM, which is `american_america.AL32UTF8`. If you use another value for `NLS_LANG`, the desired value must use the `AL32UTF8` character set. For the PostgreSQL database, use `UTF-8` encoding and a UTF-8 locale.

Make sure that you specify the desired value for the Oracle database character set before starting the HPOM installation. You can specify the character set that HPOM uses with the following command:

```
# export NLS_LANG=<value>
```

[a]ASCII is a subset of UTF-8. If only English ASCII characters will be used, it is possible to use `C` as `LANG`. However, even in this case, the usage of the UTF-8 locale is recommended. Otherwise, any multilingual data may be lost, or cause errors.

HPOM supports the Oracle database character sets listed in Table 47.

Table 47: Supported Database Character Sets and `NLS_LANG` Values

Language	Character Set	NLS_LANG Value
US English	`AL32UTF8`	`american_america.AL32UTF8`
Spanish	`AL32UTF8`	`spanish_spain.AL32UTF8`
Japanese	`AL32UTF8`	`japanese_japan.AL32UTF8`
Korean	`AL32UTF8`	`korean_korea.AL32UTF8`
Simplified Chinese	`AL32UTF8`	`simplified chinese_ china.AL32UTF8` [a]
Other	`AL32UTF8`	`american_america.AL32UTF8`

Setting Up the User Environment

All the elements in the environment that are used to access the HPOM management server must be configured to accept UTF-8 input and output. When setting up the user environment, consider the following:

- Keyboard:

  The keyboard layout (and code page) for the keyboard used to provide information sent to the HPOM management server must be able to input UTF-8 characters.

- Terminal program:

  If you use a terminal program to access the HPOM management server, you must configure it to correctly to send the user input as UTF-8. The terminal program must be able to interpret the management server's response as UTF-8 as well.

  Depending on the context, you must enable certain options, for example: by starting the terminal with special parameters, or even recompiling the terminal program with a multi-byte option.

- Fonts:

  A font capable of displaying Unicode characters must be used for the terminal that accesses the HPOM management server.

For detailed information about configuring the keyboards and terminal programs that are used to access the HPOM management server, see the system documentation for the program or operating system you are using.

[a] The space in the `NLS_LANG` variable is required.

Language Support on Managed Nodes

Table 48 and Table 49 show the languages HPOM supports for HPOM internal messages on managed nodes.

Table 48: Language Support for HPOM Internal Messages

Management Server OS	Managed Node OS	English	Japanese
HP-UX, Linux, or Sun Solaris	AIX	√	√
	HP-UX	√	√
	Linux	√	√
	Solaris	√	√
	Windows	√	√

Table 49: Language Support for HTTPS Agents Only

Management Server OS	Managed Node OS	Spanish, Korean, Simplified Chinese
HP-UX, Linux, or Sun Solaris	HP-UX	√
	Linux	√
	Solaris	√
	Windows	√

Note: Windows managed nodes use the *System* language. A *LANG* environment variable is not available.

Language Settings for Messages on Managed Nodes

The managed-node processes use the value of the locale variable to determine the language of HPOM messages. For example, if you want the managed-node processes to generate messages in Japanese, you must set the locale and language variable accordingly before you call `opcagt -start`.

Note: On the managed nodes, HPOM generates internal HPOM messages only in English and Japanese. If you have policies in any other language, make sure that the HPOM agents use the English message catalogs.

The HPOM agent processes are usually started at system boot time and inherit the system settings for the default language and character set established during the boot sequence. If, for some reason, the system default values for the language locale are not appropriate for the HPOM agent, you can use the `ovconfchg` command to change the settings and ensure that all processes belonging to HPOM (controlled by the `ovc` daemon, `ovcd`) are started in the desired locale, as follows:

```
# /opt/OV/bin/ovconfchg -ns ctrl.env -set LANG <desired_locale>
```

You cannot use this method to set the language for the HPOM agent on Windows managed nodes. On UNIX managed nodes, you must restart the HPOM agent processes using the `ovc` command, as follows:

```
# /opt/OV/bin/ovc -kill
# /opt/OV/bin/ovc -start
```

Since `ctrl.env:LANG` is set during the installation and initial setup of the HPOM management server, there shouldn't be a need to update the language setting on the management-server system later on.

Setting the Language of Messages on a Managed Node

To set the language of messages on a managed node, perform the following steps:

1. Set the locale for the HPOM agents in the system-startup script.
2. Set `START_LANG` for the locale in which you want the HPOM agent to start.
3. Restart the HPOM agents.

Locations of System Resource Files

For the location of the system resource files adapted by HPOM on all supported agent platforms, see the HP Operations Agent documentation.

Character-Set Synchronization on the HPOM Agent

The output of HPOM agent commands (for example, `opcagt -status`) uses the internal character set configured for the agent. For this reason, when the locale of the terminal window in which you execute the command is different from the internal character set of the agent, the output is not readable. If the agent has the internal UTF-8 character set, use a UTF-8 terminal window.

File-Set Requirements on Managed Nodes

Some operating systems require the installation and availability of a specific file set to convert code sets. For software requirements on all managed node platforms, see the *HPOM Software Release Notes*.

Character-Set Settings on the Managed Nodes

The character sets available on platforms supported by HPOM can differ from the character set used in the HPOM database. Consequently, when a message is generated on a managed node, it must often be converted before it can be sent to the management server and stored in the database. HPOM takes care of this conversion. If necessary, automatic character set conversions take place through HPOM managed node processes before a message is sent to the server.

Note: UTF-8 is the recommended character set, especially for environments that use multilingual characters. The HPOM database always uses UTF-8.

Character-Set Types in English or Spanish Environments

Table 50 shows the character sets for the English and Spanish languages that are supported for HPOM managed nodes.

Note: HPOM automatically sets the default of the internal agent character set to the character set supported by the lowest version of the operating system.

Table 50: Verified Character Sets on Managed Nodes (English/Spanish)

HPOM	Platform	Character Set
Management server on HP-UX, Linux, and Sun Solaris	HP-UX, Solaris	UTF-8, ISO 8859-15, ISO 8859-1, ASCII
	AIX, Linux, Solaris	UTF-8, ISO 8859-15, ISO 8859-1, ASCII
	Windows	UTF-8, multilingual ANSI Code Page 1252[a], ASCII

Character Sets in Japanese Environments

Table 51 shows the character sets for the Japanese language that are supported for HPOM managed nodes.

[a]Code Page 1252 is analogous to ISO 8859-1.

Table 51: Verified Character Sets on Managed Nodes (Japanese)

HPOM	Platform	Character Set
Management server on HP-UX, Linux, and Sun Solaris	HP-UX, Solaris	UTF-8, Shift JIS, EUC[a], ASCII
	Linux	UTF-8, EUCa, ASCII
	Windows	UTF-8, Japanese ANSI Code Page 932 [b], ASCII
	AIX	UTF-8, Shift JIS, EUCa, ASCII

External Character Sets on Managed Nodes

In mixed-language environments where the locale for the command-line shell, the HPOM agent, and the HPOM database are not consistent, you can have problems with message format and content.

All commands for HPOM managed nodes (for example, `opcmsg(1m)` and `opcmon(1m)`) use the locale setting to interpret the character set of their command-line arguments. The character set specified in the locale settings is not always the same as the character set used by the HPOM database or the character set used for message processing on the HPOM managed node. If command input is entered in a locale that is different to the locale set on the HPOM agent, the command input must be converted before it can be acted upon by any managed-node process.

Note: UTF-8 is the recommended character set, especially for environments that use multilingual characters. If UTF-8 is selected as the external character set, the internal character set of the node should also be UTF-8.

If you encounter problems with the format and content of HPOM messages, such as, garbled message text, you can change the character set on the HPOM agent, as follows:

```
# /opt/OV/bin/ovconfchg -ns eaagt -set OPC_NODE_CHARSET <charset_of_HPOM_agent>
```

In this way, you can ensure that commands such as `opcmsg` are aware of the locale set for (and used by) the HPOM agent and can provide data in the appropriate format.

Character-Set Types in English-Language Environments

Table 52 shows the relationship between the value of *LANG* set on the HPOM agent and related *external* character sets, that is, character sets used by third-party applications that communicate with HPOM using the `opcmsg` and `opcmon` interfaces in an English-language environment.

[a] 2-byte Extended UNIX Code.

[b] Code Page 932 is analogous to Shift JIS.

Table 52: External Character Sets in English/Spanish Environments

Node Platform	LANG	External Character Set
AIX	<*Lang*>.8859-15 C <*Lang*>.ISO8859-1 <*Lang*>.IBM-850 <*Lang*>.UTF-8	ISO 8859-15 ASCII ISO 8859-1 OEM Code Page 850 UTF-8
HP-UX 11.x	<*Lang*>.iso885915 <*Lang*>.iso885915@euro C <*Lang*>.iso88591 C.utf8 / <*Lang*>.utf8	ISO 8859-15 ISO 8859-15 ASCII ISO 8859-1 UTF-8
Linux	<*Lang*>@euro C <*Lang*> <*Lang*>.UTF-8	ISO 8859-15 ASCII ISO 8859-1 UTF-8
Solaris	<*Lang*>.ISO8859-15 C <*Lang*> <*Lang*>.UTF-8	ISO 8859-15 ASCII ISO 8859-1 UTF-8
Windows	*LANG* variable not available	OEM Code Page 850 OEM Code Page 437 ANSI Code Page 1252 ASCII UTF-8

The <*Lang*> variable refers to any language that is supported by the operating system. Although it is possible to specify literally any language in this field, you can receive HPOM internal messages only in a language supported by HPOM. HPOM only uses the value of *LANG* to determine the external character set.

External Character Sets in Japanese Environments

Table 53 shows the relationship between the value of *LANG* set on the HPOM agent and related *external* character sets, that is, character sets used by third-party applications that communicate with HPOM using the `opcmsg` and `opcmon` interfaces in a Japanese-language environment.

Table 53: External Character Sets (Japanese)

Node Platform	LANG	External Character Set
AIX	C	ASCII
	ja_JP	Shift JIS
	ja_JP.IBM-932	
	ja_JP.IBM-eucJP	EUC
	ja_JP.UTF-8	UTF-8
HP-UX	C	ASCII
	ja_JP.SJIS	Shift JIS
	ja_JP.eucJP	2-byte EUC
	ja_JP.utf8	UTF-8
Linux	C	ASCII
	ja_JP	EUC
	ja_JP.eucJP	EUC
	ja_JP.UTF-8	UTF-8
Solaris	C	ASCII
	ja_JP.PCK	Shift JIS
	ja	EUC
	ja_JP.UTF-8	UTF-8
Windows	*LANG* variable not available	ANSI Code Page 932
		ASCII
		UTF-8

The <*Lang*> variable refers to any language that is supported by the operating system. Although it is possible to specify literally any language in this field, you can receive HPOM internal messages only in a language supported by HPOM.

Character Sets Supported by the Log-File Encapsulator

The HPOM log-file encapsulator can monitor files with different character sets. You can specify a character set for each file monitored by HPOM. The character set can be different from the character set defined for that managed node but must be compatible.

Note: If you are using ASCII as the character set for internal processing, you must also specify ASCII as the character set for the monitored log-file messages.

ASCII is a subset of Shift JIS. You risk loss of data if you monitor Shift JIS log files by running the HPOM agent in ASCII mode.

Table 54 shows all the supported character sets for various log-file messages.

Table 54: Character Sets Supported by the Log-File Encapsulator

	Windows Nodes		HP-UX, Solaris, Linux, AIX, Nodes		Net Ware Nodes	Other Nodes
Character Set	English Spanish	Japanese	English Spanish	Japanese	English	English
ASCII	√	√	√	√	√	√
ISO 8859-15	-	-	√	-	√	√
ISO 8859-1	-	-	√	-	√	√
American EBCDIC	-	-	HP-UX	-	-	-
Multilingual OEM code page 850	√	-	AIX	-	√	-
OEM US code page 437	√	-	-	-	√	-
Multilingual ANSI code page 1252	√	-	-	-	√	-
Japanese ANSI code page 932	-	√	-	-	-	-
Shift JIS	-	-	-	√	-	-
EUC (2-byte Extended UNIX code)	-	-	-	√	-	-

Note: Code Page 932 or Code Page 1252 are the only character sets valid for the Event Log.

Character-Code Conversion in HPOM

The information in this section describes how to configure HPOM and related character sets in English- and Japanese-language environments.

Management-Server Configuration

Figure 21 shows the HPOM configuration and related character sets on an English-language management server.

Figure 21: Configuration and Related Character Sets (English)

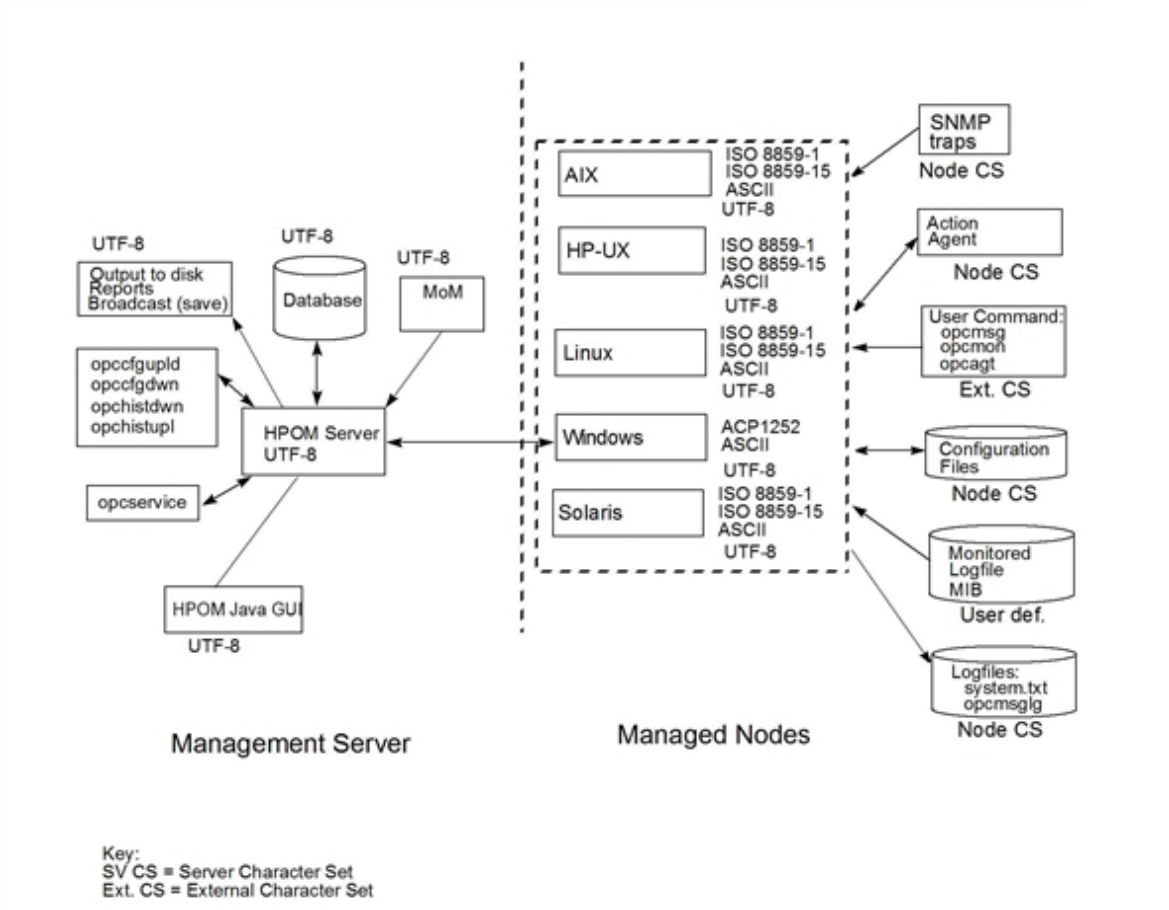
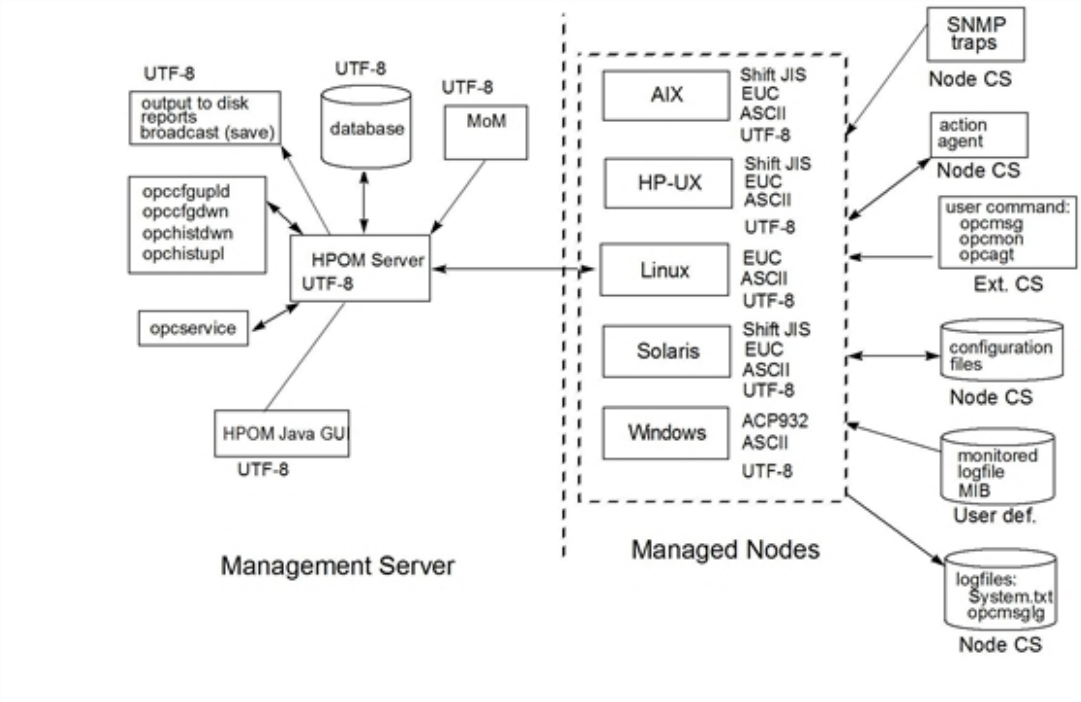


Figure 22 shows the configuration and related character sets on an HPOM management server running in a Japanese-language environment.

Figure 22: Configuration and Related Character Sets (Japanese)



Management-Server File Processing

On an English-language management server, HPOM uses the UTF-8 character set when performing the following tasks:

- Processing local log-file entries (`System.txt`), temporary queue file, and so on.
- Uploading and downloading the HPOM configuration.
- Uploading and downloading the HPOM history messages.
- Managing Service Navigator configuration with `opcservice`.

Managed-Node File Processing

In an English-language environment, HPOM processes managed node files as follows:

- SNMP events:

  Interprets incoming SNMP events in ASCII format.
- User commands:

  Converts user commands from the external character set to the node character set.
- Configuration files:

  Does not convert input for configuration files. HPOM always processes configuration files in the node processing character set.

- Local log files:

  Does not convert output for local HPOM log files. HPOM always processes the contents of log files in the node-processing character set.

- MIB processing:

  Processes MIB files in the HPOM node processing character set.

Tip: You can use a different character set for each managed node. You determine the managed-node character set by the character sets used in your environment.

In a Japanese-language environment, HPOM performs a run-time conversion for managed-node configuration files only if the HPOM agents on HP-UX, Solaris, or AIX are running with the EUC character set. HPOM does not perform a run-time conversion on the management server in a Japanese-language environment.

For example, in a Japanese-language environment where the character set for the HPOM agent on an HP-UX managed node is set to EUC and the language variable LANG is “ja_JP.SJIS”, HPOM processes the contents of managed-node files and `opcmsg` messages as follows:

Input	HPOM converts the contents of the `opcmsg` from Shift JIS to EUC.
Output	Before forwarding the message to the management server, HPOM converts the contents from EUC to UTF-8 (the database character set).

On HP-UX, you can define different character sets for different managed nodes. Define the character set most frequently used on each managed node. For example, if you use mostly monitor log files with Shift JIS characters, you should use Shift JIS for your managed nodes. If you use mostly monitor log files with ISO 8859-15 characters, you should use ISO 8859-15 for your managed nodes. When in doubt, use UTF-8.

Flexible-Management Configuration in a Japanese-Language Environment

If your management server runs in a Japanese-language environment with the character set UTF-8, you must do one of the following:

- Convert the flexible-management configuration files on the management-server from UTF-8 to EUC.

  Note: If the UTF-8 file contains the characters that are not available in EUC, problems may occur when converting the flexible-management configuration files on the management server from UTF-8 to EUC.

- Convert the managed nodes from EUC to UTF-8.

Converting Configuration Files

To convert flexible-management configuration files on the management server from UTF-8 to EUC, enter the following:

1. On HP-UX management servers:

   ```
   # /usr/bin/iconv -f utf8 -t euc <mom_orig> > <mom_new>
   ```

2. On Sun SOLARIS management servers:

   ```
   # /usr/bin/iconv -f utf8 -t eucJP <mom_orig> > <mom_new>
   ```

Note the following in the examples of the use of the `iconv` command listed above:

<*mom_orig*>	Name of the original flexible-management configuration file in UTF-8 format.
<*mom_new*>	IP address of the managed node in hexadecimal, as returned by the command `opc_ip_addr`.

Localization of Object Names

Although you can localize most of the HPOM-specific configuration, you must observe a few restrictions. For example, it is mandatory to use ASCII characters when naming the following objects:

- Managed nodes
- Files:

  Examples of files include automatic actions, scheduled actions, monitor scripts and programs, the fully qualified trouble ticket interface, and notification services.
- Monitored objects:

  For example, when using `opcmon` to forward values for objects you are monitoring with HPOM.
- Operator names:

  Operator names are used to create corresponding subdirectories and must therefore not be localized.
- Operator passwords
- HPOM administrator password

HPOM uses the name of objects (for example: the message-group name, or node-group name) as an internal identifier. For this reason, you should not localize the names of HPOM objects themselves. Instead, enter the localized string in the Label field. If a label is present, it is shown in the Java GUI instead of an internal name.

Data Download and Upload

When downloading data from the HPOM management server with the `opccfgdwn` command or uploading data with the `opcdfgupld` command, note the following important points regarding settings for language and character sets:

- Data Download:

  The `opccfgdwn` command downloads *all* data in UTF8 format. The download data is stored in the following location:

  `<user_specified_download_dir>/<user_language>.utf8`

  Even if the locale (language and character set) is set (for example, in the command shell) to `$LANG=de_DE.iso88591` when the user (or application) runs the `opccfgdwn` command, the downloaded data is, nonetheless, stored in the following location:

  `/tmp/<subdir_path>/de_DE.utf8`

  Logging data is written to an index file, which is located at the base of the download directory tree `<target directory>/<locale>` (for example, `/tmp/my_dir/de_DE.utf8`). The index file contains a reference to the character set used to encode data during the configuration download and will be used in the subsequent upload operations, only data of the same character set is uploaded.

- Data Upload:

  The `opccfgupld` command reads its data from subdirectories whose names are determined by the settings specified for the language and character set. For example, if the user calls the `opccfgupld` command in a shell with the language variable `$LANG=en_UK.utf8`, as follows:

  ```
  # opccfgupld /tmp/my_dir
  ```

  The `opccfgupld` command looks for data to upload in the following directory: `/tmp/my_dir/en_UK.utf8`

  > **Tip:** It is recommended that HPOM administrators always open command shells with a UTF8 LANG setting. UTF8 is the default setting for character-set encoding on the HPOM management server.

  The character set used at download time is checked against the default character set configured on the HPOM management server, to which you want to upload the configuration data. If necessary, the `opccfgupld` command converts data from one character set to UTF8 before starting the upload operation.

- Data Search:

  The `opccfgupld` command looks for the data to upload in locations determined by the language and character-set settings specified in the locale in which the person (or application) calling the

command is working. If there are multiple choices, the `opccfgupld` command makes the following logical assumptions:

a. If the `opccfgupld` command specifies a value for the `-upgrade` option, use the value specified. For example, `-upgrade <language_subfolder>` forces the upload of data from the specified language-specific directory, regardless of the current locale setting. This option allows you to import data to HPOM 9.x from previous versions of HPOM. The character set used for the uploaded data will be automatically detected and the data files converted to UTF8 before upload.

b. If the $LANG variable set in the shell of the user calling the `opccfgupld` command matches the language and character-set settings specified in the upload path, then use it. For example, if the caller's locale is set to `$LANG=de_DE.utf8`, and the user calls the following command:

```
# opccfgupld /tmp/my_dir
```

Then search for the data to upload in the following directory:

`/tmp/my_dir/de_DE.utf8`

c. If neither of the first two options matches, then search for the upload data in the following locations in the order specified:

i. `<base_path>/en_US.utf8`

ii. `<base_path>/C.utf8`

iii. `<base_path>/C`

iv. Try the same locations specified in the previous steps, but replace the lowercase "`utf8`" with the uppercase "`UTF-8`".

- Configuration Upload:

  If necessary, the `opccfgupld` command converts data to UTF8 before the upload operation starts.

Language Settings on the Command Line

The HPOM database expects all data to be specified in UTF8 format. However, if a user calls a command or starts a custom-written program which requires access to the application programming interface (API), on the HPOM management server, bear in mind that the command or program uses the locale set in the environment where the command or program starts.

Since no character-set conversion occurs for data accessed from command-line utilities or API calls, the locale set in the shell where the user starts the command or program must match the locale set on the HPOM management server. To be on the safe side customers should always use an UTF8 locale when working on the HPOM management server.

Troubleshooting Language Environments

The information in this section includes details of specific cases where HPOM functionality does not work as expected in international environments. For more information about installing the HP Operations management server in international environments, see the *HPOM Installation Guide for the Management Server*.

Windows Managed Nodes

In the localized versions of the Windows operating system, the user `Administrator` has been localized. Consequently, the installation of the HP Operations agent software on Windows managed nodes fails because HPOM is trying to install as user `Administrator` while the user has a different name in the Windows operating system.

To avoid problems of this kind, run the `inst.sh` script, and when asked for the user name, enter the localized name of the user.

Broadcast-Command Output

The output of the broadcast command is not always readable. This is the case if the command is run in an MS-DOS window that uses an MS-DOS code page that is different from the Windows code page.

Part VII: Maintenance

There are different processes and tools that enable the HP Operations Manager (HPOM) administrators to maintain the HPOM management server, managed nodes, and licensing and infrastructure.

For more information, see "HPOM Maintenance" on page 639.

Chapter 17: HPOM Maintenance

In this Chapter

This chapter contains information for administrators who are responsible for maintaining HPOM, maintaining the HPOM databases, and who might need to change the hostname and IP address of the management server and managed nodes.

The information in this chapter covers the following topics:

- HPOM management server:
  - "Configuration Data Download" on page 639
  - "Data Backup on the Management Server " on page 640
  - "Database Maintenance" on page 674
  - "HP Software Platform" on page 676
  - "HPOM Directories and Files" on page 677
- HPOM managed node:
  - "Managed Node Directories with Runtime Data" on page 679
  - "Location of Local Log Files" on page 679
- Licensing and infrastructure:
  - "HPOM Licenses" on page 680
  - "Host Names and IP Addresses" on page 688
  - "Host Names and IP Addresses in a Cluster Environment" on page 699

Configuration Data Download

You should download configuration data as part of your standard maintenance or backup routine. Before you make any significant changes to your HPOM configuration, make sure you download configuration data to the file system or use backup tools to back up your configuration data. For more information about backing up configuration data, see "Data Backup on the Management Server " on page 640.

You can download configuration data by using the `opccfgdwn(1m)` command. The configuration-download utility enables you to select the parts of the configuration that you want to download and save the data to flat files in the file system. For example, instead of downloading the entire configuration, you may choose to download only message-source policies or application groups.

You specify the configuration data that you want to download (for example, node groups, policies and policy groups, application groups, and so on) in the `download.dsf` file. The `download.dsf` file lists the specified objects in particular format, as illustrated in the following example:

Content and Format of the download.dsf File

```
APPLICATION_GROUP "Workspaces" ;
APPLICATION_GROUP "SiteScope SAM ADMIN" ;
MEMBER_APPLICATION_GROUP "SAM Admin2" ;
```

For more information about the syntax required in the `download.dsf` file, see the *opccfgdwn(1m)* manual page. The `download.dsf` file is required as a parameter by the `opccfgdwn(1m)` command. Unless otherwise specified, the `opccfgdwn(1m)` command writes configuration-download data in the form of a directory tree to the following location: `/var/opt/OV/share/tmp/OpC_appl`.

Note: The Administration UI writes configuration-download data to the directory: `/opt/OV/OMU/adminUI/data/clipboard/`

For more information about the parameters and options you can use to control the `opccfgdwn(1m)` command, see the *opccfgdwn(1m)* manual page. For more information about downloading configuration data using the Administration UI, see online help.

Data Backup on the Management Server

HPOM provides two methods for backing up data on the HP Operations management server:

- Offline backup:

  Use the `opcbackup_offline` utility to perform partial or full backups of data on the management server. For more information, see "Offline Backups" on page 641.

- Online backup:

  Use the `opcbackup_online` utility to perform a complete automatic backup of the database while the Java GUI and server processes are running. For more information, see "Online Backups" on page 643.

HPOM stores configuration data on the management server and on the managed nodes. If the restored configuration on the management server does not match the current configuration on a managed node, errors relating to missing instructions or incorrectly assigned policies may occur. After you have

restored a backup, you should redistribute the policies as well as the action, command and monitor scripts to all managed nodes by using the `-force` option of `opcragt`.

When recovering data, use the recover tool corresponding to the backup tool originally used to back up the data. For example, use `opcrestore_offline` to restore data backed up with `opcbackup_offline`, and use `opcrestore_online` to recover data backed up with `opbackup_online`.

With the Oracle database, you can use the `ARCHIVELOG` mode to save data automatically and periodically. Changes to data files are stored in redo log files. These redo log files are subsequently archived. For more information about the `ARCHIVELOG` mode and redo log files, see the Oracle documentation. To find out how to set up the `ARCHIVELOG` mode in HPOM, see "Backup Prerequisites" on page 645.

Offline Backups

To help you perform a backup of installation and configuration data on the HP Operations management server, HPOM provides the following scripts:

- `opcbackup_offline`:

  For more information about backing up HPOM data offline, see "opcbackup_offline Command" on page 642.
- `opcrestore_offline:`

  For more information about restoring HPOM data from an offline backup, see "opcrestore_offline Command" on page 643.

You can use the offline backup and restore tools to perform the following type of operations on the management server:

- Partial backup and restore:

  HPOM configuration data only. Includes current messages and history messages.
- Full backup and restore:

  Includes the HPOM binaries and installation defaults.

In either case, you must shut down the Java GUI and stop all HPOM processes and services.

Backing up data offline has the following advantages:

- *Oracle only:* `ARCHIVELOG` mode

  Offline backups do not require or make use of the `ARCHIVELOG` mode, which Oracle uses to save data automatically and periodically. Not using the `ARCHIVELOG` mode increases overall backup performance and requires less disk space for the backup image.
- Backup mode

  If you use the full backup mode, the HPOM binaries are included in the backup image.

Backing up data offline has the following disadvantages:

- Data recovery:

  You can recover data only to the state of the most recent full backup. With the Oracle database, it is also possible to recover part of the changes done in the database after the backup, but this is not granted.

- HPOM process and services:

  Before you start an offline backup, you must stop all HP services and the Java GUI.

For more information about the HPOM utilities that are available to help you perform offline backup and restore operations, see the *opcbackup_offline(1m)* and *opcrestore_offline(1m)* manual pages.

Running Offline Backup and Restore

Perform the offline backup by running the `opcbackup_offline` command. The `opcbackup_offline` command enables you to perform a backup of installation and configuration data as well as server binaries. During an offline backup, the HPOM database and processes are stopped. For details, see "opcbackup_offline Command" on page 642.

The `opcrestore_offline` command restores a backup created with `opcbackup_offline`. The `opcrestore_offline` command allows you to restore the complete database or corrupted HP Operations management server files. For details, see "opcrestore_offline Command" on page 643.

opcbackup_offline Command

The `opcbackup_offline` command enables you to create a backup of the whole HPOM system including the data, or just the configuration files. During the backup procedure the HP Operations server and the database are stopped.

This command accepts the following command line options:

```
opcbackup_offline [-c] [-d] [-n] [-v] [-s] [-r]
```

`-c`	If selected, only the configuration data is backed up. If no option is selected, a full backup is done.
`-d`	Use this option to add specified folders to the backup.
`-s`	*Oracle only:* This option specifies connection strings to the database on which the remote manager (RMAN) will perform the backup. The format of the string is as follows: *<user>/<password>@<server>/<dbname>* or *<user>/<password>@<server>/<dbalias>*

	where *<user>* can be, for example, `system`. If not specified, the current HPOM database instance is used.
`-r`	Use this option to specify the location where the database manager (`RMAN` for Oracle or `pg_dumpall` for PostgreSQL) will store the backup of the database. In an Oracle environment, the specified location is relative to the database server. In a PostgreSQL environment, the specified location is relative to the HP Operations management server. It is preferred that the directory is new and empty.
`-n`	No interaction with the command.
`-v`	Verbose mode.

For more information about the parameters and options available with the offline-backup utilities, see the *opcbackup_offline(1m)* and *opcrestore_offline(1m)* manual pages.

opcrestore_offline Command

To restore a backup image made with `opcbackup_offline` tool, run the `opcrestore_offline` command in a command shell and answer the prompts for information that the command displays:

```
# /opt/OV/bin/OpC/opcrestore_offline
```

For more information about command options and parameters, see the *opcrestore_offline(1m)* manual page.

Online Backups

To help you perform a complete automatic backup of the database while the Java GUI and server processes are running, HPOM provides the following scripts:

- `opcbackup_online`:

  For more information about backing up HPOM data online, see "opcbackup_online Command" on page 649.

- `opcrestore_online`:

  For more information about restoring HPOM data from an online backup, see "opcrestore_online Command" on page 649.

You can manage the online backups using cron jobs or through scheduled HPOM actions.

Online backups have the following advantages:

- Java GUI:

  There is no need to close the Java GUI, although OVW actions are not possible for a short time.

- Processes and services:

  HPOM server processes, Java GUI services, trouble ticket services, and notification services remain fully operational.

- Database:

  Oracle only: Partial recovery of the database is possible.

  For example, you could recover the Oracle database as follows:

  - All data backed up to a given time.
  - Only individual damaged table spaces.

Online backups have the following disadvantages:

- *Oracle only:* `ARCHIVELOG` mode

  Oracle `ARCHIVELOG` mode must be enabled:

  - Reduces overall performance.
  - Requires more disk space.

- Binaries

  No binaries are backed up.

For more information about the HPOM utilities that are available to help you perform online backup and restore operations, see the *opcbackup_online(1m)* and *opcrestore_online(1m)* manual pages.

Temporary Files in Online Backups

Temporary files (for example, queue files) are excluded from online backups. When a backup starts, the Java GUI pops up a notification window and some HPOM maps remain blocked for the duration of the backup. If a task cannot be completed before the backup starts, the task remains idle until the backup operation completes. After the backup completes, the suspended task resumes.

Archive Log Mode in Oracle

The scripts provided by HPOM for automated backups use the online backup method from Oracle, which requires the database run in the `ARCHIVELOG` mode. The Oracle `ARCHIVELOG` mode is not the default setting for the Oracle database; you must configure `ARCHIVELOG` mode manually.

In `ARCHIVELOG` mode, Oracle stores any changes to data files between full backups in numbered redo log files. The redo log files are used in the event of a shutdown to restore a configuration from the most recent, full backup. For details, see the Oracle product documentation.

For more information about enabling `ARCHIVELOG` mode, see "Backup Prerequisites" on page 645.

Backup in an Oracle RAC Environment

HPOM supports online backup in an Oracle Real Application Cluster (RAC) environment. The procedure for backing up a HPOM system running in the Oracle RAC environment is similar to backing up a system with a remote database. However, some additional preparation steps are needed. For the RAC specific requirements and procedures, see "Online Backup and Restore in an Oracle RAC Environment" on page 651.

Note: Only online backup scripts are supported with the Oracle RAC environment.

Backup Notification Tools

The `opcwall(1)` command line utility enables you to notify all running Java GUI instances of an imminent automated backup.

You can use the following parameters and options with the `opcwall` command:

`opcwall {-user <user_name>} <Message Text>`

`-user`	Single user to whom you want to send a message. If not specified, all operators receive the message.
`<user_name>`	Name of the operator you want to receive the message.
`<Message Text>`	Text of the message you want the operator to see.

Backup Prerequisites

Before performing either online or offline backups, note the following prerequisites. Before you start, it is recommended to create a pair of folders with all rights included. Backup scripts for both types of backup (online and offline) are at the following location: `/opt/OV/bin/OpC`.

If you want to perform online backups, use the following commands:

- `opcbackup_online`
- `opcrestore_online`

If you want to perform offline backups, use the following commands:

- `opcbackup_offline`
- `opcrestore_offline`

If you want to perform online backup in an Oracle RAC environment, see the RAC-specific prerequisites in "Online Backup and Restore in an Oracle RAC Environment" on page 651.

Backup Prerequisites for the Oracle Database

To create either online or offline backups, ensure that the following prerequisites are met:

1. Configure access to the remote database tool (RMAN) for the internal database user SYSTEM.

   The password for remote-database access is normally set during the HPOM installation process and stored in encrypted form in the .opcdbrem.sec file. If the .opcdbrem.sec file does not exist, you must create it manually. To write the existing RMAN password for database user SYSTEM in encrypted form to the .opcdbrem.sec file, run the following commands:

   ```
   # RMAN_PASSWD=manager
   # export RMAN_PASSWD
   # /opt/OV/bin/OpC/opcdbpwd -rpr
   # unset RMAN_PASSWD
   ```

   This operation only stores the password in the encrypted file and does not change the password of the database SYSTEM user in the database. If you want to change the password of the database SYSTEM user, run the following command:

   ```
   SQL> alter user system identified by <new-password>;
   ```

2. In cluster environments, shut down the monitor on the Oracle resource group before setting the database to ARCHIVELOG mode. Otherwise, the monitor will detect that the database is not available and a failover will occur. Run the following command:

   ```
   # /opt/OV/lbin/ovharg -monitor ov-oracle disable
   ```

3. Set the database to ARCHIVELOG mode for online backup (for offline backup, the ARCHIVELOG setting is optional) as follows:

   a. Depending on your system, choose one of the following:

      - On Unix and Linux systems:

        Log on as the oracle user by running the following command:

        ```
        # su - oracle
        ```

      - On Windows systems:

        Move to the <*ORACLE_HOME*>\bin directory logged on as the Oracle owner.

   b. Run the following commands:

      ```
      $ sqlplus /nolog
      SQL> conn / as sysdba
      SQL> shutdown immediate
      SQL> startup mount
      SQL> alter database archivelog;
      SQL> alter database open;
      ```

      To check if the database is set, use the following command:

      ```
      SQL> archive log list;
      SQL> exit
      ```

4. In cluster environments, restart the monitoring of the Oracle resource group by running the following command:

   ```
   # /opt/OV/lbin/ovharg -monitor ov-oracle enable
   ```

5. Grant SYSTEM user permission to use the remote-database management tool, RMAN:

   Note that the action described in this step occurs automatically if the following is true:

   - Database creation:

     The HPOM database has been created by HPOM (not manually) and it is not a remote database.

   - Database configuration:

     During setup of the HPOM database, you replied in the affirmative to the question "Configure the database for remote login?".

   To grant SYSTEM user access to the remote database manager (RMAN), perform the following steps:

   a. Create the password file by using the following commands:

      ```
      $ sqlplus /nolog
      SQL> conn / as sysdba
      SQL> alter system set remote_login_passwordfile=exclusive scope=spfile;
      SQL> shutdown immediate
      SQL> startup
      SQL> exit
      ```

      ```
      $ orapwd file=<ORACLE_HOME>/dbs/orapw<ORACLE_SID> password=<SYSTEM_password>
      ```

      For example:

      ```
      $ orapwd file=/opt/oracle/product/11.2.0/dbs/orapwopenview password=manager
      ```

      Note: You can safely ignore the following error message:

      ```
      OPW-00005: File with same name exists - please delete or rename
      ```

   b. Grant permissions to the SYSTEM user by using the following command:

      ```
      $ sqlplus /nolog
      ```

      ```
      SQL> conn / as sysdba
      Connected.
      ```

      ```
      SQL> grant SYSDBA to SYSTEM;
      Grant succeeded.
      ```

   c. Check the permissions for the SYSTEM user:

      ```
      SQL> select * from v$pwfile_users;

      USERNAME SYSDB SYSOP
      ------------------------------ ----- -----
      SYS TRUE TRUE
      ```

```
SYSTEM TRUE FALSE
SQL> exit
```

6. Make a note of the Oracle database ID (DBID).

   The DBID is a code that identifies an Oracle database. In the event of a catastrophic failure, some manual steps might be required to recover the database. In this case, it is essential to know the database ID. To retrieve the database ID, run the following command:

   ```
   $ sqlplus /nolog
   SQL> conn / as sysdba
   SQL> select dbid from v$database;
   SQL> exit
   ```

Note: The RMAN version must be compatible with the Oracle server.

Backup Prerequisites for the PostgreSQL Database

To create either online or offline backups, ensure that the following prerequisites are met:

1. Make sure that the database is configured to allow access for the administrator user inside the database cluster or the server (that is, the `DB DBA user`). To do this, check the `<cluster_dir>/pg_hba.conf` and `.pgpass` files. For details, see the PostgreSQL documentation.
2. The password for the `DB DBA user` is set during the HPOM installation process and stored in encrypted form in the `.opcdbrem.sec` file. If the `.opcdbrem.sec` file does not exist, you must create it manually by running the following commands:

   ```
   # RMAN_PASSWD=<DB_DBA_password>
   # export RMAN_PASSWD
   # /opt/OV/bin/OpC/opcdbpwd -rpr
   # unset RMAN_PASSWD
   ```

   This operation only stores the provided password in encrypted form and does not modify the actual password inside the database.

Running Online Backup and Restore

Perform the online backup by running the `opcbackup_online` command. The `opcbackup_online` command enables you to perform a backup of installation and configuration data while the HPOM database is online. You do not have to stop the database to perform an online backup. For details, see "opcbackup_online Command" on page 649.

The `opcrestore_online` command restores a backup created with `opcbackup_online`. The `opcrestore_online` command allows you to restore the complete Oracle database or corrupted files. You can restore the database either to the state of the backup or, in the case of the Oracle database, to the most recent state. For details, see "opcrestore_online Command" on page 649.

opcbackup_online Command

The following list describes the parameters and options you can use with the `opcbackup_online` command:

`opcbackup_online [-c] [-r] [-s] [-v]`

`-c`	Oracle only: String to use to connect to the database where the remote manager (RMAN) will perform the backup. The format of the string is as follows: *<user>/<password>@<server>/<dbname>* or *<user>/<password>@<server>/<dbalias>* where *<user>* can be, for example, `system`. If no connection string is specified, the current HPOM database instance is used.
`-r`	Location where the database manager (`RMAN` for Oracle or `pg_dumpall` for PostgreSQL) will store the backup of the database. In an Oracle environment, the specified location is relative to the database server. In a PostgreSQL environment, the specified location is relative to the HP Operations management server. It is preferred that the directory is new and empty. *Oracle only:* A network mount can only be used for this as long as the mount path is the same for all database nodes, if more than one.
`-s`	Location of the HPOM data backup folder.
`-v`	Verbose mode.

opcrestore_online Command

Before running `opcrestore_online`, make sure that `/opt/OV/bin` is included in your `PATH`.

Note: Before starting, `opcrestore_online` verifies that no HP Software or integrated processes are running.

This command accepts the following command-line options:

`opcrestore_online [-c] [-s] [(-b |-l)] [-v]`

`-c`	*Oracle only:* Connection string to the database on which the remote manager (RMAN) will perform the backup.

`-s`	Folder where HPOM backup files are stored.
`-l`	*Oracle only:* Restore the latest recoverable state of the database.
`-b`	*Oracle only:* Restore the data until the time of the most recent backup.
`-v`	Verbose mode.

Alternative Backup Methods

Depending on your environment or needs, you may want to use a different approach to backing up your HPOM data. In this case, for example, you can use one of the following backup methods:

- "Oracle Online Backup" on page 650
- "Full File System Backup" on page 650

Oracle Online Backup

The Oracle online backup method enables you to perform an online backup of the database by using RMAN. With this backup method, most of the operational data is stored and can be recovered in case of a database failure. However, this backup does not include any of the HPOM configuration files, binaries, or downloaded history messages. Therefore, you cannot recover HPOM configuration files or binaries if they become damaged. In this case, the configuration files must be reinstalled or recreated manually. To avoid this scenario, you can combine the Oracle online backup with a full file system offline backup or an HPOM offline backup.

Full File System Backup

The full file system backup method enables you to make a copy of the whole file system of the HP Operations management server when the HP Operations processes and the database are down. By doing this you have a complete and consistent copy of the HPOM data and configuration files. Unlike the HPOM offline backup scripts, this method allows you to back up the operating system package inventory (`SD` on HP-UX, `pkginfo` on Solaris, and `RPM` on Linux) containing a list of the installed operating system, HPOM, and third-party filesets and patches.

When using the full file system backup in cluster environments, consider the following:

- If you need to stop the database or processes, make sure that you disable monitoring.
- Shared disks are not mounted if the cluster HARG is completely stopped.

To maintain access to the shared disks for backup, it is recommended that you follow these steps:

1. Put the HARG into maintenance mode (by doing this, you also disable monitoring).
2. Exit all GUIs, and then stop all agent and server processes.

3. Perform the offline backup.
4. Restart the agent and server processes.
5. Enable monitoring of the HARG again.

Backup Considerations

With all backup methods (HPOM backups and alternative backup methods), take into account the following considerations:

- HPOM stores configuration data on the management server and the managed nodes. If the restored configuration on the management server does not match the current configuration on a managed node, errors relating to missing instructions or incorrectly assigned policies may occur. After you restore a backup, you should redistribute the policies as well as the action, command, and monitor scripts to all managed nodes by using the `opcragt` command with the `-force` option.
- Messages are stored in the database and therefore they can increase the required amount of disk space when you back up the database. To reduce the backup size, use the `opcackmsg` and `opchistdwn` commands periodically for acknowledging and downloading the messages. After that you can store downloaded messages separately from the main backup.
- *Oracle only:* Oracle redo logs and control files can be mirrored. Having multiple copies of redo logs and control files enables you to recover a damaged file by copying it over from one of the other locations. RMAN also allows backing up important configuration files (control files, redo logs, `SPFILE`). Using these options increases the chances of a successful restore in a wider range of database failures.
- Each backup method has advantages and disadvantages. Therefore, the best backup strategy can be a combination of several backup methods. For example, you can combine a monthly full file system backup (before or after major changes are performed on the system) with a weekly online backup and a daily backup of all configuration data with `opccfgdwn`.

Online Backup and Restore in an Oracle RAC Environment

Caution: Only online backup is supported in an Oracle RAC environment.

This section contains RAC-specific requirements and steps for performing backup and restore. A sample RAC environment, which is referred to in this section, consists of two nodes, `node1(-vip)` and `node2(-vip)`. Each node has an instance, `GRID1` or `GRID2`, both these instances use a common database, `openview`. To connect to this database from the HPOM server, `ov_net` is used as an alias.

During the database backup, RMAN connects to the database by using the `ov_net` alias, which means that the backup is performed by either of the nodes. Before restore, however, all instances of the

HPOM database except for one are stopped, and a connection to this node is established to perform the restore procedure.

For more information on configuring nodes in the Oracle RAC environment, see the *HPOM Installation Guide for the Management Server*.

Prerequisites

Before starting an online backup and restore, make sure that the following steps are performed to configure your RAC environment:

1. Configure the Oracle Net Listener files.

   During the restore, the database is shut down. If the instances are registered dynamically with the listener, it is not possible to connect to them to restore and reactivate the database. Because of this, you must add a static entry to the `listener.ora` file for each node.

   For example, add the following lines to the `listener.ora` file of `node1`:

   ```
   (SID_LIST_LISTENER_NODE1 =
     (SID_LIST =
       (SID_DESC =
         (SID_NAME = GRID1)
         (GLOBAL_DBNAME = openview)
         (ORACLE_HOME = /opt/oracle/product/11.1.0)
     )
   )
   ```

   Restart the listener on each node for the changes to take effect.

2. Set up an alias in the `tnsnames.ora` file.

   During the restore, all instances except one are down. To connect to the remaining instance, you must set up an alias to connect to a particular node.

   On the HP Operations management server, add the following lines to `tnsnames.ora`:

   ```
   NODE_1 =
   (DESCRIPTION =
     (ADDRESS = (PROTOCOL = TCP) (HOST = node1-vip) (PORT = 1521))
     (CONNECT_DATA =
        (SID = GRID1)
     )
   )

   NODE_2 =
   (DESCRIPTION =
     (ADDRESS = (PROTOCOL = TCP) (HOST = node2-vip) (PORT = 1521))
     (CONNECT_DATA =
       (SID = GRID2)
     )
   )
   ```

3. Set the database in ARCHIVELOG mode.

   Because only an online backup is supported with RAC, the database must be set in ARCHIVELOG mode. To enable ARCHIVELOG mode, follow these steps:

   a. If the management server is installed, stop the processes by running the following command:

      ```
      # ovc -kill
      ```

   b. In HPOM cluster environments, shut down the monitor on the Oracle resource group before setting the database to ARCHIVELOG mode. Otherwise, the monitor will detect that the database is not available and a failover will occur.

      Run the following command:

      ```
      # /opt/OV/lbin/ovharg -monitor ov-oracle disable
      ```

   c. Stop all instances except one (for example, GRID1) by running the following command from any of the RAC nodes:

      ```
      # su - oracle
      $ srvctl stop instance -d openview -i GRID2[,GRID3[,GRID4…]]
      ```

   d. Set the database in ARCHIVELOG mode as follows:

      Note: The archived logs must be stored in a shared location (the NFS mount with a mount path common to all RAC nodes or ASM storage). This can be done by changing the log_archive_dest or the log_archive_dest_n database parameter.

      i. On the RAC node that has a running instance (for example, instance GRID1), do the following:

         ```
         # su - oracle
         $ sqlplus /nolog
         SQL> conn / as sysdba
         SQL> alter system set log_archive_format="T%TS%SR%R.ARC" scope=spfile;
         SQL> alter system set log_archive_dest='+DATA/' scope=spfile;
         SQL> shutdown immediate;
         SQL> startup mount
         SQL> alter database archivelog;
         SQL> alter database open;
         SQL> exit
         ```

      ii. On each of the RAC nodes, do the following:

         ```
         # su - oracle
         $ sqlplus /nolog
         SQL> conn / as sysdba
         SQL> startup
         SQL> exit
         ```

   e. Ensure that all instances are started by running the following command:

      ```
      # su - oracle
      $ srvctl start database -d openview
      ```

f. Check if all database instances are running as follows:

```
# srvctl status database -d openview
```

g. Check the mode of the database on each of the RAC nodes as follows:

```
# su - oracle
$ sqlplus /nolog
SQL> conn / as sysdba
SQL> select log_mode from v$database;
LOG_MODE
------------
ARCHIVELOG
SQL> exit
```

h. In HPOM cluster environments, restart the monitoring of the Oracle resource group by running the following command:

```
# /opt/OV/lbin/ovharg -monitor ov-oracle enable
```

i. Start the management server by running the following command:

```
# ovc -start
```

4. Grant permissions to the SYSTEM user to connect remotely to the database as SYSDBA.

a. If the management server is installed, stop the processes by running the following command:

```
# ovc -kill
```

b. Enable RMAN to connect remotely to both RAC nodes. Run the following commands on each of the RAC nodes:

```
# su - oracle
$ sqlplus /nolog
SQL> conn / as sysdba
SQL> alter system set remote_login_passwordfile=exclusive scope=spfile;
SQL> shutdown immediate
SQL> startup
SQL> exit
$ orapwd file=<ORACLE_HOME>/dbs/orapw<ORACLE_SID> password=<SYSTEM_password>
```

For example:

```
$ orapwd file=/opt/oracle/product/11.1.0/dbs/orapwopenview password=manager
$ sqlplus /nolog
SQL> conn / as sysdba
Connected.
SQL> grant SYSDBA to SYSTEM;
Grant succeeded.
```

c. Check the permissions for the SYSTEM user:

```
SQL> select * from v$pwfile_users;

USERNAME SYSDB SYSOP SYSAS
------------------------------ ----- ----- -----
```

```
SYS TRUE TRUE  FALSE
SYSTEM TRUE FALSE FALSE
SQL> exit
```

Note: You can safely ignore the following error message:

```
OPW-00005: File with same name exists - please delete or rename
```

d. Connect to each instance from the HP Operations management server by using the listener aliases created in Step 1 on page 652:

```
# su - oracle
$ rman target system/manager@NODE_1
$ rman target system/manager@NODE_2
...
```

e. To store the encrypted password, run the following commands on the management server:

```
# RMAN_PASSWD=manager
# export RMAN_PASSWD
# /opt/OV/bin/OpC/opcdbpwd -rpr
# unset RMAN_PASSWD
```

f. Ensure that all instances and services are up by running the following commands on one of the nodes:

```
# su - oracle
$ srvctl start database -d openview
$ srvctl status database -d openview
Instance GRID1 is running on node <node1>
Instance GRID2 is running on node <node2>
```

g. Restart the HPOM services by running the following command:

```
# /opt/OV/bin/ovc -start
```

5. Use shared storage.

   Because the backup can be performed by any of the RAC nodes, you must specify a shared location where to store the backup data (for example, an NFS folder that is mounted on the same place for all nodes or in ASM storage). The same is valid for shared elements of the database, such as the archived redo logs.

Performing Backup in an Oracle RAC Environment

When your environment meets the requirements described in "Prerequisites" on page 652, you can start an online backup in the Oracle RAC environment. To perform a backup, use the same procedure as described in "Running Online Backup and Restore" on page 648.

Performing Restore in an Oracle RAC Environment

To perform the restore procedure in an Oracle RAC environment, follow these steps:

1. Stop the HPOM services by running the following command:

   ```
   # /opt/OV/bin/ovc -stop
   ```

2. Stop all the instances except for one by running the following command from any of the instances:

   ```
   # su - oracle
   $ srvctl stop instance -d openview
   ```

3. Run the `opcrestore_online` tool with the `-c` option that is used for specifying the connection string for the active node:

   ```
   # opcrestore_online -c <RMAN_connection_string> -s <backup_location>
   ```

   For example:

   ```
   # opcrestore_online -c system/manager@NODE_1 -s /hpombackup
   ```

4. After the restore procedure is complete, restart the instances by running the following command:

   ```
   # su - oracle
   $ srvctl start instance -d openview
   ```

5. Restart the HPOM services by running the following command:

   ```
   # /opt/OV/bin/ovc -start
   ```

Data Recovery After an Automatic Backup

Automatic backup scripts only make a backup of configuration data and dynamic data. If binaries or static configuration files are lost, you must recover them first before restoring the database.

You can recover binaries or static configuration files in one of the following ways:

- Restore a full offline backup:

  Restore a full offline backup of the complete system, which was taken using `opcbackup_offline` with the `full` option.

- Reinstall HPOM:

  When recovering binaries or static configuration files, choose the reinstall method as the last option. Reinstalling server packages can lead to the loss of custom configuration data.

Reinstalling Management Server Packages

To reinstall all packages that are installed during the HPOM management-server installation process, perform the following steps:

1. Stop all HPOM server components by running the following command:

   ```
   # /opt/OV/bin/ovc -kill
   ```

2. Reinstall all packages by running the `ovoinstall` command, as follows:

   ```
   # /opt/OV/bin/OpC/install/ovoinstall -force -skip_setup_check -pkgdir <package_repository>
   ```

In this instance, *<package_repository>* is a full path to the location of the repository containing the HPOM management server packages.

Caution: If the server is already configured, do not continue with the configuration.

3. Start all server components:

```
# /opt/OV/bin/ovc -start
```

Restoring a Database to the State of Its Latest Backup

Restoring the HPOM database to its state at the time of the last backup only requires the data contained in the backup.

Restoring an Oracle database in this way leaves the database in an inconsistent state because the latest state of the database is not restored. In addition, Oracle log numbers are reset in the control files and in the online redo logs. After successfully completing this kind of restore, you will need to create a new backup.

Restoring an Oracle Database to Its Latest State

Restoring the Oracle database to the last known state is more complicated than restoring the database to its state at the time of the last backup. Restoring the database to its last known state requires not only the data contained in the backup but also data on the system itself, namely: online redo logs and the archive logs written since the last backup.

Note: Restoring the database to its last known state can introduce inconsistencies between the configuration files (restored to the state of the backup) and the data in the database (restored to the last known state between the last backup and the time the restore operation starts).

Before you attempt to recover the database to its last known state, make sure you perform the following checks:

1. Check the control files:

   All control files must exist. Normally, control files are mirrored and backed up. If one of the control file still exists, it can be copied from one location to the other. Control files can also be extracted from the backup. However, this should be done by an Oracle database administrator (DBA). Note that scripts can only restore to the latest state if all control files exist.

2. Check the redo log files:

   All online redo log files must exist. Online redo log files are backed up and can be mirrored. If one of the online redo log files in a log group still exists, it can be copied to the other locations. This should be done by an Oracle DBA. Note that scripts can only restore to the latest state if all redo log files exist.

3. Check the Oracle log number:

   Oracle log numbers are reset in the control files and in the online redo logs during a backup. The Oracle log number must not have been reset since the backup.

4. Check the archived redo logs:

   All archived redo logs made since the backup must still exist and be available.

5. Check the status of HPOM Users:

   No HPOM users should have been modified since the backup, which modifies files in the file system.

6. Check the event-correlation (ECS) policies:

   No ECS policies should have been added since the backup.

Removing HPOM Queue Files

HPOM queue files are neither backed up with the automated backup scripts nor deleted during the restore. In addition, the messages in the queue files at the time of the backup are not in the database and are processed only when the HPOM processes are next restarted.

If corrupt queue files prevent the server processes from being started, remove the queue files.

To remove the queue files, follow these steps:

1. Stop all HP Operations server processes:

   ```
   # /opt/OV/bin/OpC/opcsv -stop
   ```

2. Remove a selected temporary file or all temporary files:

   ```
   # rm -f /var/opt/OV/share/tmp/OpC/mgmt_sv/*
   ```

3. Restart the HP Operations server processes:

   ```
   # /opt/OV/bin/OpC/opcsv -start
   ```

Manual Recovery of an HPOM Database

For a local database, the HPOM restore scripts are able to restore a database backup to the state of the backup even if control files or the complete database are missing.

Manual Recovery of an Oracle Database

For a remote database (also if the database was on a different system when the backup was taken, like in a decoupled cluster setup), the database may be damaged in such a way that it cannot be automatically recovered by the HPOM scripts. These cases may also include either a loss of one or more `controlfile` copies or `SPFILE`. However, the HPOM scripts keep separate the copies of these files in the database backup folders. For example:

- OPENVIEW_DBID**2654967530**_22-04-09_10.29_ctrl_6_684844242_1

  for controlfile

- OPENVIEW_DBID**2654967530**_22-04-09_10.29_cfg_5_684844240_1

  for spfile

An additional copy of controlfile exists. It is kept by Oracle in its autobackup location, which is by default $ORACLE_HOME/dbs:

OPENVIEW_c-**2654967530**-20090422-01_ctrlautobackup

The files at these locations are exact copies of the missing files and can be used with RMAN to recover the database.

Make sure that you consider the following before performing a manual recovery of the HPOM database:

- The listener process must be up and running. Otherwise RMAN cannot connect to the instance.
- In certain cases DBID of the database may be needed. This ID code must be written down during the backup preparation steps by running the following query:

  ```
  SQL> select dbid from v$database;
  ```

  Note: DBID is also part of the file names in the database backup folders.

- If the Oracle instance has not been shut down yet, shut it down before making an attempt to recover it:

  ```
  # su – oracle
  $ sqlplus system/<password>@<ov_net> as sysdba
  SQL> shutdown abort
  SQL> exit
  ```

To start the manual recovery procedure, enter the following command:

```
# su – oracle
$ rman target system/<password>@<ov_net>
RMAN> SET DBID <DBID>
RMAN> startup nomount
```

The Oracle database instance should be running at this point, after which you can try to recover the damaged file or the damaged files:

- If the SPFILE is damaged, enter the following command:

  ```
  RMAN> restore SPFILE from
  ‘<full path of the ‘cfg’ file in the backup folder>’;
  ```

  Append to ‘*<PATH>*’ at the end of the above command if you want to create a copy of the SPFILE at the location specified in *<PATH>*.

Note: You can safely ignore the RMAN-06564 error that is displayed if you run the suggested command on an `SPFILE` that is not corrupted.

- If `controlfile` is damaged, enter the following command:

  ```
  RMAN> restore controlfile from
  '<full path of the 'ctrl' file in the backup folder>';
  ```

  If you fail to recover the damaged `controlfile` by running the suggested command, Oracle can attempt to recover it from a copy made during an automatic backup over the course of the last year, which is the maximum time allowed. To restore the `controlfile` from an automatic backup copy, enter the following command:

  ```
  RMAN> restore controlfile from autobackup maxdays 366;
  ```

After successfully recovering the `SPFILE` and `controlfile`, start the database recovery process using the following commands:

```
RMAN> startup mount
RMAN> restore database;
RMAN> recover database;
RMAN> alter database open resetlogs;
RMAN> exit
```

The database should be up and running at this point. It is recommended to make a new backup after you have checked that everything is in order.

Note: If you want to run an HPOM restore script after manually restoring the database, use the option for restoring data to the most recent possible time (for example, `opcrestore_online -l`). The `offline-restore` tool, `opcrestore_offline`, prompts you for information during the restore process.

Manual Recovery of a PostgreSQL Database

Whenever an offline or online backup is performed, the `pg_dumpall` command is run. This command stores an SQL dump file of the whole database cluster, including users, permissions, and all the databases within it.

In addition, it is possible to use the SQL dump file directly to recover only the database or make changes before uploading the data again. However, keep in mind that this may cause data corruption and downtime.

To perform a manual SQL recovery, follow these steps:

1. If the existing database cluster is not up or is damaged, remove and recreate it either manually or by using the `psqlcluster` and `psqlsetup` tools.
2. Locate the SQL dump file inside the backup directory or tar file. The name of this file is given in the

following format:

`OMUBak_<year_month_day>_<hour_min_sec>.gz`

3. Decompress the SQL dump file to get it in plain text:

   `gunzip <dumpfile>`

4. Edit the file according to your needs.
5. Restore the database cluster with the modified SQL dump file by running the following commands:

   ```
   # su - <OS_DBA_user>
   $ <PSQL_bin_dir>/psql -U <DB_DBA_user> -h <hostname> -p <port> -f <dump_file>
   ```

Event Storm Filter

The Event Storm Filter (ESF) program provides a mechanism to filter HPOM events when an event storm (that is, a large number of events in a short time) is detected. Because events generate messages that specify a type of these events, the number of generated messages grows with an increase in the number of events. For detailed information about events, event storms, and messages, see the *HPOM Concepts Guide*.

Not only do event storms overload the management server, but they may represent a potential problem when it comes to trouble ticketing and notification systems. For example, in an operating environment of a delivery center, having the ESF program may be of a crucial importance because it ensures that an event storm generated at a single customer site (or multiple sites) does not negatively affect trouble ticketing and notification systems.

To manage the ESF, use the `opcesf.sh` tool that you can find at the following location:

`/opt/OV/bin/OpC/utils/`

For detailed information about the `opcesf.sh` tool usage and options, see the *opcesf.sh* manual page.

When testing and troubleshooting, you can use a log file analysis that represents a useful methodology for understanding all the aspects of the ESF process. For these purposes, use the `opcesf.log` file that can be found at the following location:

`/var/opt/OV/share/tmp/OpC/esf/`

Note: The `opcesf.log` file is deleted each time the ESF process is restarted.

Keep in mind that reinitializing the ESF by running the `opcesf.sh -init` command does not clear the `opcesf.log` file.

ESF Process Modes

The ESF process runs in two different modes at the same time—the log mode and the gate mode. The log mode is used to define which information is to be written to a log file, while the gate mode is used to specify whether message filtering should be limited to one single filter (one gate) or multiple filters (gates).

When defining the log mode and the gate mode for the ESF process, make sure that you choose both kinds of modes. The following log and gate modes are available:

- **Log modes:**
  - Normal (default)

    This mode is used for the normal operation. Error messages and messages that cause a storm are logged to the `opcesf.log` file.

    To run the ESF in this mode, set the `OPC_ESF_LOG_LEVEL` configuration variable to `NORMAL`.

  - Verbose

    This mode is used for debugging and troubleshooting. All messages are logged to the `opcesf.log` file.

    To run the ESF in this mode, set the `OPC_ESF_LOG_LEVEL` configuration variable to `VERBOSE`.

  - Silent

    In this mode, only error messages are logged to the `opcesf.log` file.

    To run the ESF in this mode, set the `OPC_ESF_LOG_LEVEL` configuration variable to `SILENT`.

- **Gate modes:**
  - Event matches one gate

    To run the ESF process in the mode in which an event matches only one gate, set the `OPC_ESF_ONE_GATE` configuration variable to `TRUE`.

  - Event matches multiple gates (default)

    To run the ESF process in the mode in which an event matches multiple gates, set the `OPC_ESF_ONE_GATE` configuration variable to `FALSE`.

To change the mode, you can use the `opcesf.sh` tool. For example, to change the mode to verbose, run the following command:

```
/opt/OV/bin/OpC/utils/opcesf.sh -verbose
```

For detailed information about configuration variables, see the *HPOM Server Configuration Variables* document.

Enabling the Event Storm Filter

The ESF can be installed and enabled during the installation of the HPOM software on the management server. For details, see the *HPOM Installation Guide for the Management Server*.

Note: If you decided not to enable the ESF during the management server installation, you can do it later by running the following command:

```
/opt/OV/bin/OpC/utils/opcesf.sh -enable
```

By running this command, you run the `opcesf` process with a certain initial configuration. For example, this means that in case of a message storm with a rate of ten or more messages in two minutes from any managed node, the ESF functionality detects it as such and sends a critical message to the browser. All the messages that come from that managed node after the tenth message are discarded until a defined period of time elapses. For details, see "Customizing the ESF Flood Gate Configuration File" on page 664.

If you plan to use some other programs that connect to the MSI as well (for example, `opcecm`), make sure that the values you specify in the `msiconf` file meet the following requirements:

- `HealthCheck` has a lower value than `opcesf`.
- `opcesf` has a lower value than some other program.

For example:

```
HealthCheck 1
opcesf 11
opcecm 31
```

Note: The `msiconf` file is located in the following directory:

```
/etc/opt/OV/share/conf/OpC/mgmt_sv/
```

For more information, see the *msiconf* manual page.

Disabling the Event Storm Filter

If you do not need the ESF functionality, you can disable it by running the following command:

```
/opt/OV/bin/OpC/utils/opcesf.sh -disable
```

Configuring the Event Storm Filter

When configuring the ESF, you can perform the following operations:

- "Customizing the ESF Flood Gate Configuration File" on page 664
- "Customizing the ESF Configuration File" on page 672
- "Creating a Customer Information File" on page 673

Customizing the ESF Flood Gate Configuration File

The default ESF flood gate configuration file, `flood_gates.conf`, defines how the ESF filters HPOM events when an event storm is detected. This configuration file can be found at the following location:

`/etc/opt/OV/share/conf/OpC/mgmt_sv/esf/`

You can customize the default ESF flood gate configuration file according to your preferences by defining parameters for each flood gate. The parameters describe criteria for each type of storm and specify how the following will appear:

- Storm-detected messages, which are generated when a storm is detected.
- Storm-ended messages, which are generated when a storm ends.

Caution: After modifying the ESF flood gate configuration file, you must reinitialize the `opcesf` process to reload the changes. To do this, run the following command:

```
/opt/OV/bin/OpC/utils/opcesf.sh -init
```

Table 55 shows which parameters you can use to customize the ESF flood gate configuration file according to your preferences.

Table 55: ESF Flood Gate Configuration File Parameters

Parameter	Value	Description
`GateName` *(required)*	`value` Text string (spaces allowed) representing the name of the flood gate.	Represents the label for the ESF to use when referring to event storms detected by this gate.
`Customer` *(required)*	`!`, `%`, `value%`, or `value` `!` Matches messages that use this parameter as a key. For example, if the `Customer` field is set to `!`, the `Rate` number of events must occur in a specified period of time (`Period`) on the nodes of a specified customer for a storm to be reported. The events that occur on the nodes	Defines how the `Customer` parameter is checked for this gate. An event must match all required gate parameters (that is, `Customer`, `Node`, `Object`, `Application`, `MsgGroup`, `MsgType`, `MsgText`, and `TT`) to be considered a match for a flood gate.

ESF Flood Gate Configuration File Parameters, continued

Parameter	Value	Description
	of other customers are ignored. `%` Matches messages regardless of the value in this field. `value%` Matches messages if the parameter begins with a string value. `value` Matches messages if a string value is completely matched.	
`Node` *(required)*	See "Customer (required) ".	Defines how the `Node` parameter is checked for this gate. NOTE: Node matching is case insensitive.
`Object` *(required)*	See "Customer (required) ".	Defines how the `Object` parameter is checked for this gate.
`Application` *(required)*	See "Customer (required) ".	Defines how the `Application` parameter is checked for this gate.
`MsgGroup` *(required)*	See "Customer (required) ".	Defines how the `MsgGroup` parameter is checked for this gate.
`MsgType` *(required)*	See "Customer (required) ".	Defines how the `MsgType` parameter is checked for this gate.
`MsgText` *(required)*	See "Customer (required) ".	Defines how the `MsgText` parameter is checked for this gate.
`TT` *(required)*	`!`, `%`, `0`, or `1` `!` Matches messages that use this parameter as a key. `%` Matches messages regardless of the value in this field. `0` Matches messages that do not have the trouble ticket flag set. `1` Matches messages that have the	Defines how the `TT` parameter is checked for this gate.

ESF Flood Gate Configuration File Parameters, continued

Parameter	Value	Description
	trouble ticket flag set.	
`Rate` *(required)*	`1` or higher	Specifies how many events must happen in a specified period of time (`Period`) for events to be considered a storm.
`Period` *(required)*	`1` or higher	Specifies a time frame (in minutes) in which a number of events greater than or equal to the specified `Rate` must occur so that events are considered a storm.
`WarnAgainPeriod` *(optional)*	`1` or higher	Specifies in how many minutes another critical warning message must be sent if a storm remains open. For example, `WarnAgainPeriod=60` means that an additional warning is sent every hour until the storm ends.
`ExcludeMsgGroup` *(optional)*	Any message group name	Specifies a message group value for incoming events. The result is that an event does not match the gate even if all other parameters match. For example, `ExcludeMsgGroup=HC` means that all events with the `HC` message group are sent back to the MSI without matching any gate. You can specify any number of `ExcludeMsgGroup` values, each in a separate line.
`ExcludeNode` *(optional)*	Any node name (fully qualified)	See "ExcludeMsgGroup" (applies to a node value).
`ExcludeSeverity` *(optional)*	`NORMAL`, `MINOR`, `MAJOR`, `WARNING`, or `CRITICAL`	See "ExcludeMsgGroup" (applies to a severity value).

ESF Flood Gate Configuration File Parameters, continued

Parameter	Value	Description
`ExcludeObject` *(optional)*	Any object name	See "ExcludeMsgGroup" (applies to an object value).
`ExcludeApplication` *(optional)*	Any application name	See "ExcludeMsgGroup" (applies to an application value).
`Log` *(optional)*	`1` (log) or `0` (do not log) Default: `0`	Specifies whether to log suppressed messages to a log file during a storm.
`Annotate` *(optional)*	`1` (annotate) or `0` (do not annotate) Default: `0`	Specifies whether to add filtered messages as annotations to a storm-detected message.
`CreateTT` *(optional)*	`1` (create a trouble ticket) or `0` (do not create a trouble ticket) Default: `0`	Specifies whether to create a trouble ticket for a storm-detected message.
`WarnMsgGroup` *(optional)*	Any message group name The default value is the message group of the first message in the storm if matching is made based on a message group, or `OpC` if matching is not based on a message group.	Defines a message group for a storm-detected message sent by the ESF.
`WarnObject` *(optional)*	Default: `Event Storm`	Defines an object for a storm-detected message.
`WarnMsgText` *(optional)*	The default message is the following: `Flood Gate <gatename> has detected a message storm. The ! fields specified in the gate match (<value of matched fields>).`	Defines message text for a storm-detected message. This message text is preceded or followed by the information about the fields that matched the gate and whether the storm is logged to a file or as annotations.
`WarnMsgTextPos` *(optional)*	`FIRST` or `LAST` `FIRST` The custom message text (`WarnMsgText`) precedes additional	Defines where to place custom message text for a storm-detected message.

ESF Flood Gate Configuration File Parameters, continued

Parameter	Value	Description
	information. LAST The custom message text follows additional information. Default: FIRST	NOTE: If WarnMsgText is not specified, WarnMsgTextPos is ignored.
WarnInfoText *(optional)*	Additional information about the message that generated the message storm.	Enables customization of ESF messages.
WarnSeverity *(optional)*	NORMAL, MINOR, MAJOR, WARNING, or CRITICAL Default: CRITICAL	Defines severity for a storm-detected message.
WarnNode *(optional)*	LOCALHOST or SERVER LOCALHOST The node that generated a first storm message is used as a node for a storm warning message. SERVER The HP Operations management server name is used instead. Default: SERVER	Defines the value of a node parameter for a storm-detected message.
OpAction *(optional)*	SHOW_LOG or any command SHOW_LOG: This is translated to the cat <*logfile*> command where <*logfile*> is the file that contains all logged events for a specified storm. The OpActionNode parameter must be set to SERVER because the log files are kept on the management server. The Log parameter must be set to 1 (true). Default: no action is performed.	Defines an operator-initiated action to be set for a storm-detected message.
OpActionNode *(optional)*	LOCALHOST or SERVER Default: LOCALHOST	Specifies the node on which OpAction is to be performed.

ESF Flood Gate Configuration File Parameters, continued

Parameter	Value	Description
AutoAction *(optional)*	SHOW_LOG or any command SHOW_LOG: This is translated to the cat <*logfile*> command where <*logfile*> is the file that contains all logged events for a specified storm. The AutoActionNode parameter must be set to SERVER because the log files are kept on the management server. The Log parameter must be set to 1 (true). Default: No action is performed.	Defines an automatic action to be performed for a storm-detected message.
AutoActionNode *(optional)*	LOCALHOST or SERVER Default: LOCALHOST	Specifies the node on which AutoAction is to be performed.
CloseMsg *(optional)*	1 (send a storm-ended message) or 0 (do not send a storm-ended message) Default: 0	Specifies whether a message is generated when a storm ends.
CloseAction *(optional)*	/opt/OV/bin/OpC/opcackmsg -u opc_adm or any other command that takes a message ID as the last argument. The message ID is automatically added by the opcesf process. The value should be omitted or set to none if no action is desired at the end of the storm.	Specifies an automatic action that is called by the storm-ended message. If this action is successful (that is, exits with 0), the storm-ended message is automatically acknowledged.
EndMsgGroup *(optional)*	Any message group name Default: the message group is the same as the message group of the storm-detected message (that is, the value of the gate field if matching is based on it, or OpC if matching is	Defines a message group for a storm-ended message.

ESF Flood Gate Configuration File Parameters, continued

Parameter	Value	Description
	not based on it).	
`EndCreateTT` *(optional)*	`1` (create a trouble ticket) or `0` (do not create a trouble ticket) Default: `0`	Specifies whether to create a trouble ticket for a storm-ended message.
`MsgKeyPrefix` *(optional)*	`message key prefix` Text string representing the message key prefix.	Assigns message keys for each flood gate entry. Message key is set to *<message key prefix>*:*<node name>*:START, for message indicating that message storm is detected. Message key is set to *<message key prefix>*:*<node name>*:END, for message indicating the end of message storm, if corresponding option `CloseMsg` is enabled in the flood gate. Here, *<message key prefix>* is taken from the `flood_gates.conf` file. *<node name>* is the name of the node where the message storm originated. **Note:** Message key relation is set to ^*<message key prefix>*:*<node name>*:<*>$, to indicate that end of message storm automatically acknowledges start of message storm and vice-versa.
`CMA` *(optional)*	*name=value* In this instance, *name* is the name of the custom message attribute and *value* is the value of the CMA.	Assigns custom message attributes to ESF messages.

Note: If both the HC component and the ESF component are enabled, it is highly recommended to exclude the `HealthCheck` message group (`ExcludeMsgGroup=HC`), so that all events with the `HealthCheck` message group are sent back to the MSI without matching any gate. Otherwise, the `HC Alive` messages coming from the monitored nodes might cause unwanted event storms and the HC status might be incorrect.

You can generate an ESF status report in HTML form by running the following command:

```
/opt/OV/bin/OpC/utils/opcesf.sh -status
```

Examples of How ESF Messages are Composed

The following table shows different ways in which ESF messages are composed:

Table 56: Examples of ESF Message Composition

Parameter Setting	Example	Description
`WarnMsgText` is not set	`Flood Gate 'Catch ALL w/o TT' has detected a message storm. The "!" fields specified in the gate match (Node=abc.hp.com, Object=o, Application=a)`	The default ESF message is displayed, as the `WarnMsgText` parameter is not set. Additional information is displayed after the default message.
`WarnMsgText` set to: `This is my ESF warning` `WarnMsgTextPos` not set `WarnInfoText` not set	`This is my ESF warning (Node=abc.hp.com, Object=o, Application=a) Logfile: None Annotations: No`	The custom ESF message is set using `WarnMsgText`. Additional information is displayed after the custom ESF message, as `WarnMsgTextPos` is not set, and it takes the default value of `FIRST`.
`WarnMsgText` set to: `This is my ESF warning` `WarnMsgTextPos` set to `LAST` `WarnInfoText` not set	`(Node=abc.hp.com, Object=o, Application=a) This is my ESF warning Logfile: None Annotations: No`	The custom ESF message is displayed after the additional information, as `WarnMsgTextPos` is set to `LAST`.
`WarnMsgText` set to: `This is my ESF warning` `WarnMsgTextPos` not set `WarnInfoText` set to	`This is my ESF warning (informational message)`	The custom ESF message is displayed before `informational message` set using `WarnInfoText`.

Examples of ESF Message Composition, continued

`informational message`		
`WarnMsgText` set to: `This is my ESF warning` `WarnMsgTextPos` set to `LAST` `WarnInfoText` set to `informational message`	`(informational message) This is my ESF warning`	The custom ESF message is displayed after `informational message` set using `WarnInfoText`, as `WarnMsgTextPos` is set to `LAST`.
`WarnMsgText` not set `WarnInfoText` set to `informational message`	`Flood Gate 'Catch ALL w/o TT' has detected a message storm. The "!" fields specified in the gate match (informational message)`	The default ESF message is displayed, but the additional information is replaced by `informational message` set using `WarnInfoText`.

Customizing the ESF Configuration File

The ESF configuration file, `opcesf.conf`, determines how the `opcesf` tool works. You can find this file at the following location:

`/etc/opt/OV/share/conf/OpC/mgmt_sv/esf/`

Table 57 shows which parameters you can use to customize the ESF configuration file according to your preferences.

Table 57: ESF Configuration File Parameters

Parameter	Value	Description
`CUSTOMER_INFO_FILE`	Path to the customer information file Default: `customer_info_file`	The customer information file contains relationships between nodes and customers (when HPOM manages nodes from different customers). If the file does not begin with the "`/`" character, the default path is `$OV_CONF/OpC/mgmt_sv/`. Otherwise, the full path must be specified.
`MSI_BUF_SIZE`	Number of messages Default: `0`	Determines the maximum number of messages that HPOM buffers for the `opcesf` tool. If the number is exceeded, `opcesf` returns a -42 error

ESF Configuration File Parameters, continued

Parameter	Value	Description
		code when reading from the Serial MSI. The maximum number of messages should depend on the frequency of the messages. If you want `opcesf` to run continuously, set the value to `0`.
`LOGFILE_SIZE`	Size in MB Default: `4`	Determines the maximum size of the `opcesf.log` file. If it is exceeded, `opcesf.log` is renamed to `opcesf.log.<DATE>`.
`CMA_EVENTSOURCE`	Value for the `EventSource` CMA Default: `MS_OVO`	Defines the CMA with the name `EventSource` and the default value `MS_OVO`. This CMA is added to all ESF alarm messages.
`CMA_EVENTTYPE`	Value for the `EventType` CMA Default: `hpom`	Defines the CMA with the name `EventType` and the default value `hpom`. This CMA is added to all ESF alarm messages.

Creating a Customer Information File

If you use the ESF to manage multiple customers and you want to filter events based on customers, create a customer information file to map customer names with the list of nodes that are managed by that customer.

The default location of the customer information file (`customer_info_file`) is the following:

`/etc/opt/OV/share/conf/OpC/mgmt_sv/esf/`

If you want to specify another location, you must edit the `CUSTOMER_INFO_FILE` parameter in the following file:

`/etc/opt/OV/share/conf/OpC/mgmt_sv/esf/opcesf.conf`

The syntax of the customer information file is as follows:

```
Customer=Customer Name
node_name_1 node_name_2… node_name_n
Customer=Another Customer Name
another_node_name_1
```

Caution: When specifying node names, make sure that they are all listed in one line and separated by spaces. The node names must also match the node names as specified in HPOM (fully qualified host names).

After modifying the ESF general configuration file and/or the ESF customer information file, you must reinitialize the `opcesf` process to reload the changes. To do this, run the following command:

```
/opt/OV/bin/OpC/utils/opcesf.sh -init
```

Database Maintenance

To ensure that your HPOM database runs efficiently, you should perform the following tasks periodically:

- History-message browser:

  If a very large number of messages have been produced (for example, by an inappropriately configured policy), operators may find that the `Message Browser` takes a long time to open. In this case, as user root, use the command-line utilities `opcack` or `opcackmsg` to acknowledge these messages and move them to the history database. For details, see the *opcack(1m)* and *opcackmsg (1m)* manual pages.

  The tool `/opt/OV/bin/OpC/opcdbmsgmv` moves all messages that are marked as acknowledged to the history-message tables in the database, where they are retained with little or no negative effect on operational tasks. Although automatically started every two hours by the HPOM control manager, `opcdbmsgmv` may also be called manually for troubleshooting purposes.

- History messages:

  Download history messages by using the `opchistdwn` command line tool. To restore previously backed up history messages, see the *opchistupl(1m)* or *opcaudupl(1m)* manual page, and *opchistdwn(1m)* manual page for downloading history messages.

- HPOM configuration:

  Back up the HPOM configuration regularly. For details, see "Data Backup on the Management Server " on page 640.

- Disk space:

  The HPOM database files automatically consume the extra disk space required to cope with any growth in the backup image. If a disk runs out of space, you can use other disks to add additional tablespaces and files. For details, see the database product information.

- *Oracle only:* Audit files:

  Every time a user runs the command `connect internal`, Oracle adds an audit file to the directory *$ORACLE_HOME*`/rdbms/audit`. Because the monitor policy `mondbfile` runs the `connect internal`

command roughly every ten minute, you should review the files in this directory regularly and, if necessary, remove them.

Database Configuration on Multiple Disks

Configuring the database on multiple disks enables you to increase reliability of the database server by keeping copies of important database files in different media.

With the Oracle database configuration, you can move one or more Oracle control files or online redo logs to another disk. For details, see "Oracle Database Configuration on Multiple Disks" on page 675.

The PostgreSQL database includes a set of files that cannot be mirrored, but it is possible to move them to another disk by creating soft links. For example, you can create a soft link to the `pg_control` file or the `pg_xlog` directory inside the PostgreSQL cluster. However, it is recommended that you use backup or synchronization instead.

Oracle Database Configuration on Multiple Disks

Although using the Oracle archive-log mode helps to reduce the loss of data after backing up and restoring a database, Oracle offers additional ways to avoid data loss in the unlikely event that a disk fails.

If you can access more than one disk, you should review the following configuration tips. Use the information provided when implementing similar scenarios in your own HPOM environment.

Moving Oracle Control Files to a Second Disk

To move one or more Oracle control files to the second disk, follow these steps:

1. Create the directories on the second disk:

   ```
   # mkdir -p /u02/oradata/om
   # chown oracle:dba /u02/oradata/om
   ```

2. Shut down the database.
3. Move selected control files to a directory on the second disk, for example, from disk `/u01` to disk `/u02`:

   ```
   # mv /u01/oradata/om/control03.ctl /u02/oradata/om/control03.ctl
   ```

4. Modify the control file names in the following file:

   `$ORACLE_HOME/dbs/init${ORACLE_SID}.ora`

   Example of old control file names:

   ```
   control_files = (/u01/oradata/om/control01.ctl,
                    /u01/oradata/om/control02.ctl,
                    /u01/oradata/om/control03.ctl)
   ```

Example of new control file names:

```
control_files = (/u01/oradata/om/control01.ctl,
                 /u01/oradata/om/control02.ctl,
                 /u02/oradata/om/control03.ctl)
```

5. Restart the database.

Creating a Set of Mirrored Online Redo Logs

You can create a second (or even third) set of mirrored, online redo logs on the second (or even a third) disk. HPOM installs Oracle in such a way that, by default, it has three redo log groups, each containing one member.

The following procedure shows how to create a second set of redo log files in the directory. `/u02/oradata/om`. Modify the directory names (and repeat the steps) as required.

To create a second set of redo log files, perform the following steps:

1. Create the directories on the second disk.

   Example:

   ```
   # mkdir -p /u02/oradata/om
   # chown oracle:dba /u02/oradata/om
   ```

2. As user `oracle`, enter the following:

   ```
   # sqlplus /nolog
   SQL>connect / as sysdba
   alter database add logfile member ‘/u02/oradata/om/redo01.log’ to group 1;
   alter database add logfile member ‘/u02/oradata/om/redo02.log’ to group 2;
   alter database add logfile member ‘/u02/oradata/om/redo03.log’ to group 3;
   exit
   ```

HP Software Platform

To maintain the HP Software platform, periodically verify that the trap-daemon log file `trapd.log` has not grown too large. A large trap-daemon log off can reduce the performance of HPOM.

A backup file of `trapd.log` is also provided in the following location:

`/var/opt/OV/log/trapd.log.old`

If you no longer need the entries logged in the `trapd.log` file, erase the log file, which you can find in the following location:

`/var/opt/OV/log/trapd.log`

For details about system maintenance in HP NNMi, see the NNMi documentation.

HPOM Directories and Files

To maintain HPOM directories and files, bear in mind the following guidelines:

- Management server directory:

  Important runtime data is contained in the directory `/var/opt/OV/share/tmp/OpC/mgmt_sv`. Do not clean up this directory unless you are unable to use another solution or there are too many unprocessed and old messages.

- Software installation file:

  If you no longer need the information appended to log files during software installation, update, and removal, you should backup and then erase the following log file:

  `/var/opt/OV/log/OpC/mgmt_sv/install.log`.

  The `inst_err.log` and `inst_sum.log` log files do not continuously grow because they are generated for each HPOM software installation, update, or removal.

- Error log file:

  You should back up and then erase the HPOM error and warning log file and its backups (for HTTPS-based managed nodes):

  - Plain text:

    `/var/opt/OV/log/System.txt`

  - Binary:

    `/var/opt/OV/log/System.bin`

  HPOM uses an automatic backup log-file mechanism having up to ten files. To save disk space, if the `System.txt` log-file size is greater than one (1) MB, HPOM automatically performs the following clean-up actions:

  - Moves `System.txt.008` to `System.txt.009`
  - Moves `System.txt.007` to `System.txt.008`
  - Moves `System.txt.006` to `System.txt.007`
  - Moves `System.txt.005` to `System.txt.006`
  - Moves `System.txt.004` to `System.txt.005`
  - Moves `System.txt.003` to `System.txt.004`
  - Moves `System.txt.002` to `System.txt.003`

- Moves `System.txt.001` to `System.txt.002`
- Moves `System.txt` to `System.txt.001`

HPOM Managed Nodes

On the managed nodes, you should periodically back up, and then erase, local HPOM log files (and their backups). HPOM uses 90% of the specified log directory size for local message logging, and 10% for error and warning logging. HPOM also uses an automatic backup mechanism for the log files (four on UNIX and Solaris).

For example, the configured size of a UNIX log directory is 10 MB. The size of a UNIX log directory is allocated in the following way:

- Message logging:

  HPOM allocates 9 MB for local message logging. Given that there are four log files, if the `opcmsglg` file size is greater than 2.25 MB, HPOM does the following:

  - Moves `opcmsgl2` to `opcmsgl3`
  - Moves `opcmsgl1` to `opcmsgl2`
  - Moves `opcmsglg` to `opcmsgl1`

- Error and warning message logging:

  HPOM allocates 1 MB for local error and warning message logging. If the `System.txt` (on HTTPS-based managed nodes) file size is greater than 1 MB, HPOM does the following:

  - Moves `System.txt.008` to `System.txt.009`
  - Moves `System.txt.007` to `System.txt.008`
  - Moves `System.txt.006` to `System.txt.007`
  - Moves `System.txt.005` to `System.txt.006`
  - Moves `System.txt.004` to `System.txt.005`
  - Moves `System.txt.003` to `System.txt.004`
  - Moves `System.txt.002` to `System.txt.003`
  - Moves `System.txt.001` to `System.txt.002`
  - Moves `System.txt` to `System.txt.001`

Managed Node Directories with Runtime Data

Table 58 shows the managed node directories that contain important runtime data.

Table 58: Managed Node Directories Containing Runtime Data

HPOM	Operating System on the Managed Node	Directories Containing Runtime Data
Management server on: • HP-UX • Linux • Solaris	AIX	/var/lpp/OV/tmp/OpC /var/lpp/OV/tmp/OpC/bin /var/lpp/OV/tmp/OpC/conf
	HP-UX 11.x Linux Solaris	/var/opt/OV/tmp/OpC /var/opt/OV/tmp/OpC/bin /var/opt/OV/tmp/OpC/conf
	Windows	\usr\OV\tmp\OpC\<node> \usr\OV\tmp\OpC\bin\intel \usr\OV\tmp\OpC\conf\<node>

Unless there is no alternative, or if there are too many unprocessed and old messages, do not clean up these directories.

Location of Local Log Files

Table 59 shows where local log files reside on HTTPS-based managed nodes running the HP-UX 10.x, 11.x, or Windows operating systems.

Table 59: Local Log Files on HP-UX 10.x/11.x and Windows HTTPS-based Managed Nodes

Log File	Windows	HP-UX 10.x and 11.x
Default log-file path	\Program Files\HP\OpenView\data\log	/var/opt/OV/log
HPOM errors and warnings	System.txt System.txt.(001-003)	System.txt System.txt.(001-003)
HPOM messages	opcmsglg opcmsgl(1-3)	opcmsglg opcmsgl(1-3)

Table 60 shows where local log files reside on AIX HTTPS-based managed nodes.

Table 60: Local Log Files on AIX HTTPS-based Managed Nodes

Log File	AIX
Default log-file path	/var/opt/OV/log/
HPOM errors/warnings	`System.txt`
HPOM messages	`System.txt`

Table 61 shows where local log files reside on other UNIX managed nodes.

Table 61: Local Log Files on Other UNIX HTTPS-based Managed Nodes

Log File	Linux and Solaris
Default log-file path	`/var/opt/OV/log/System.txt`
HPOM errors/warnings	`System.txt` `System.txt.(001-003)`
HPOM messages	`opcmsglg` `opcmsg (1-3)`

HPOM Licenses

The HPOM licensing component is a utility that enables you to manage the deployment and registration of HPOM licenses. The licensing component checks if licenses are available and finds out if objects that require a license have the appropriate license. The HPOM licensing component includes the `ovolicense` tool, which is a license management utility that enables you to add, enable, or disable license passwords, check a license status, and generate a license report.

Note: With HPOM 9.xx, licensing information is no longer stored in the database, but sent by the agent to its primary manager once a day. Licenses are required on the license manager only in a backup server environment.

Licensing Component Configuration

The licensing component automatically checks the validity of deployed HPOM licenses once a day. If the licensing component discovers any problems, it sends a notification message to the HPOM message browser and a designated email recipient. In case there are no problems, a notification message is sent to the designated email recipient. The HPOM licensing component has the following prerequisites:

- Unix `mailx` utility:

  Ensures the delivery of email notifications to a designated email recipient (for example, the license administrator).

- Configuration parameters:

  Define where to send notification messages concerning licensing problems and what the scope and contents of the license report should be.

  For more information about viewing and changing the configuration settings for the licensing component, see "License Component Configuration Parameters " on page 681.

Email Utility

The UNIX utility program `mailx` must be correctly configured to ensure that the HPOM licensing component can send license status messages to the license administrator. The availability of `mailx` has no effect on the functionality of HP Operations Manager but enables it to send license notification messages.

License Component Configuration Parameters

The HPOM licensing component uses parameters to configure the generation of notification messages and reports. The parameters must be adapted to the requirements of the user environment in which the licensing component runs. You can use the following parameters to configure the HPOM licensing component:

- `LicenseAdminEmailAddress`

  Email address of the person responsible for HPOM license management or the person monitoring the HPOM license status. The default setting is `root@<local_long_hostname>`. Change the setting as soon as possible to reflect the needs and configuration of your environment.

- `Content`

  Level of detail for license reports, which list the number and status of available, installed, and used licenses for each installed HPOM component. The level can be set to Summarized (default) or Detailed. Detailed license reports can be very long if there is a large number of configured nodes. To reduce the length of licensing reports, set the content level to Summarized. For more information about the contents of license reports, see "License Reports" on page 682.

Setting License Component Configuration Parameters

Licensing configuration parameters are set in the `[opr.el]` configuration name space. To set the configuration parameters for the licensing component, use the `ovconfchg` command, for example:

```
# /opt/OV/bin/ovconfchg -ovrg server -ns opr.el -set LicenseAdminEmailAddress
license_admin@company.com -set Content Summarized
```

The parameters set in this example ensure that a notification message is sent automatically to the `license_admin` user at the designated email address after a daily HPOM licensing check.

Licensing reports generated by the `ovolicense` utility are in the short, summarized form.

Note that the `ovconfchg` command also enables you to change the configuration settings using a text editor such as `vi` or `emacs`. For more information about the `ovconfchg` command, see the *ovconfchg(1)* manual page.

License Reports

The licensing component enables you to generate a license report that shows you which licenses are needed, how many are installed, and how many are in use. The report also indicates how many licenses are still available and for how long they are valid.

OM License Reporter

The `omlicreporter` command is a license reporting tool that enables you to check the status and availability of HPOM licenses as well as to generate HTML license reports.

The OM License Reporter creates the following reports:

- Feature License Report

  Shows the status of all HPOM features and licenses. This report indicates how many licenses are installed and how many licenses are already in use. The status indicates the overall license status of a feature.
- License Password Report

  Shows a detailed list of all installed HPOM license passwords. This list enables you to check which license passwords are installed and which features are enabled by each license password.

The syntax of the `omlicreporter` command is the following:

`omlicreporter [<feature_report_file> <pwd_report_file>]`

In this command, `<feature_report_file>` is the target for the OM Feature License Report and `<pwd_report_file>` is the target for the OM License Password Report.

Unless otherwise specified, the `omlicreporter` tool writes the HTML license reports into the following files:

`/opt/OV/www/htdocs/ito/OMLicenseFeatureReport.html`

`/opt/OV/www/htdocs/ito/OMLicensePasswordReport.html`

The HTML license report can be accessed with a web browser at the following locations:

`http://<management_server>:8081/ITO/OMFeatureLicenseReport.html`

`http://<management_server>:8081/ITO/OMLicensePasswordReport.html`

`https://<management_server>:8444/ITO/OMLicensePasswordReport.html`

ovolicense Tool

The `ovolicense` command is a license management tool that enables you to check the status and availability of HPOM licenses and generate license reports. The `ovolicense` command also enables you to add, enable, or disable license passwords.

Synopsis

```
ovolicense
   -c|-check -u|-feature <plugin_id> -p|-product <product>
   -e|-email -p|-product <product> <report_options>
   -g|-gui -a|-category <category>
   -h|-help
   -i|-install -a|-category <category> [-f|-file <pwd_file>]
   -l|-list [-a|-category <category>]
   -m|-mappings
   -q|-request -a|-category <category>
   -r|-report -p|-product <product> <report_options>
   -s|-status -p|-product <product>
```

For more information about the options you can use to specify the format and content of the reports generated by the licensing component, see "Report options" on page 683.

Report options

```
[-xml|-text] [-detailed] [-out <file>] [-quiet]
```

Unless otherwise specified, `ovolicense` generates a license report in summarized text form. You can use the following options to change the report format and content:

`-xml`	Creates a license report in XML format (instead of the default text format). The XML format is useful if further processing of the report data is required. Target Connector history data is part of the XML report. Note that XML reports are always detailed regardless of the setting specified with the `-detailed` option.
`-text`	Creates the license report in text format. This is the default setting.
`-detailed`	Creates an extended report containing additional information about license about all configured nodes. This can make the report very long, if the number of licensed nodes is high.
`-out <file>`	Writes the report output to a specified file name and location.
`-quiet`	Suppresses comments and progress information during the generation of the

	license report.

Options

Note that since some `ovolicense` functions use Java, the `JAVA_HOME` variable must be set to a valid runtime value. The GUI features available with `ovolicense` require Java and an X11 display. You can use the following options with the `ovolicense` command:

`-help`	Displays a list of available options for the `ovolicense` command.
`-mappings`	Shows which product components are registered for licensing and to which category they belong, for example, HPOM (for HPOM on UNIX).
`-install -category HPOM [-file <password_file>]`	Enables the installation of new license passwords stored in a file. By default, all license passwords in the specified file are installed during the operation. If no file is specified, a pop-up window prompts you to specify the file containing the license passwords and enables you to select a subset of passwords within the file, if necessary.
`-request -category HPOM`	Opens a GUI window allowing you to request and install license passwords belonging to an order number.
`-gui -category HPOM`	Opens the license report GUI without any specific functionality selected.
`-status -product HPOM`	Reports the license status of all registered license components for a product. For HP Operations Manager, the product is always "`HPOM`".
`-email -product HPOM [<report_options>]`	Generates a license report for HPOM license registrations on the basis of the report options and sends it to the email address specified in the `LicenseAdminEmailAddress` configuration parameter. By default, the email address is set during initial configuration to user `root` on the machine hosting the HPOM management server. Use the `ovconfchg` command to change the default address to the email address used by your license administrator. For information about changing this setting, see the *ovconfchg(1)* manual page. For HPOM on UNIX, the product name is HPOM. If you do not use the report options to specify a particular format, `ovolicense` generates a summarized report in text format by default and attaches the report to the email.
`-report -product HPOM [<report_options>]`	Generates a license report on the basis of the report options and prints it to standard out in a console.

Example Text License Report

"License Report" on page 686 shows an HPOM text license report that displays details of all licensed HPOM components. The heading of the report displays information about the installed version of HP Operations Manager as well as the current patch level. The body of the report shows the number of installed, used, and available licenses for each HPOM component. The final part of the report shows an overview of the configuration parameters used by the `ovolicense` command to send message notifications and generate the license report.

Text license reports comprise the following sections:

- Agent count:

  Virtual license that is not part of the HPOM product and does not represent an installed license. It is used to summarize all agent licenses on all agent tiers. The Agent Count section of the example report displayed in "License Report" on page 686 shows that there is an insufficient number of installed HPOM agent licenses.

- HP Operations management server:

  Status of the HP Operations management server license, for example: `OK`.

- HP Operations Manager tier agent:

  Status of the HP Operations agent license. The number of used licenses for the Desktop Agent and Tier 0 to Tier 4 Agent is always zero because the agent tier cannot be detected and the agent license requirement cannot be assigned to the correct license type. Use Agent Count to count and summarize license status.

- Unpatched nodes:

  Number of nodes that are not using up-to-date HPOM agent software. Licenses required for the node will only be reported by the HPOM agent software if it is up to date.

- Unreachable nodes:

  Number of nodes that sent license and node details but have not refreshed their data for more than 14 days.

Figure 23: License Report

```
HP Operations Manager License status report of Tue Jan 20 17:22:00 2009
================================================================

HP Operations Manager Server Information:
=========================================
Product Name       : HP Operations Manager for Unix
Version            : 09.00.000
Patch Level        : 09.00.000
Management server  : omuserver
Total of mgd nodes : 86

HP Operations Manager License Summary:
======================================

Agent Count
-----------------------------------------------------------
  Installed Licenses      : 62
  Used Licenses           : 86
  Available Licenses      : -24

  ===========================================================
  CRITICAL: 24 'Agent Count' licenses are missing.
  Please acquire at least 24 'Agent Count' licenses.
  ===========================================================

HP Operations Manager Target Connector
-----------------------------------------------------------
  Installed Licenses      : 1
  Used Licenses           : 0
  Available Licenses      : 1

HP Operations Manager Server
-----------------------------------------------------------
  Installed Licenses      : 1
  Used Licenses           : 1
  Available Licenses      : 0

HP Operations Manager Tier 0 Agent
-----------------------------------------------------------
  Installed Licenses      : 10
  Used Licenses           : 0
  Available Licenses      : 10

  Number of unpatched nodes   : 17
  Number of unreachable nodes : 1

Configuration Parameters:
=========================

  License Manager Mail Address : license_admin@company.com
  License Report Content       : Summarized
  License Warning Severity     : Major
  Disable License Warnings     : FALSE
```

Unregistered Components

The following example shows a license report for an object that is not registered for licensing purposes. The `ovolicense` tool can produce this type of report when a configuration from one HP Operations

management server is uploaded onto a different HPOM server which does not have the same components or SPIs installed. The report indicates that the first management server has license requirements that are different to the requirements on the second management server.

To solve this problem, the components or SPIs listed as unregistered must either be installed on all HPOM management servers that share a configuration or removed from the HPOM nodes whose configuration is shared by the management servers.

License Report: Unregistered Component

```
------------------------------------------------------------
* Not Registered: ‘noregspi’
------------------------------------------------------------
  Installed Licenses      : 0
  Used Licenses           : 10
  Available Licenses      : -10

  ============================================================
  CRITICAL: 10 licenses with the plugin ID ‘noregspi’
  are used by one or more nodes, but the according component
  is either not installed or is corrupt. Please install the
  missing component and make sure that a sufficient number of
  licenses is installed.
  ============================================================
```

The following example shows the critical message sent to the HPOM Message Browser once a day if there is a mismatch between installed components (for example, a Smart Plug-in) and registered licenses.

Licensing Error Message in the Message Browser

```
Can't check license status because of missing ID mapping file.
Error: '(oprel-124) ID mapping file does not exist: (oprel-123)
Can't find ID mapping file '/opt/OV/misc/EL/registration/<plugin>.xml' for plug-
in'<plugin>'.
```

Reporting Licenses in a Server Pooling Environment

The license report displays the license count for managed nodes that have a physical management server as the license manager.

In a server pooling environment, the license manager may be set to a virtual node name, and the node license is not counted on either physical management server.

To enable license count of virtual nodes, run the following command:

```
/opt/OV/bin/ovconfchg -ovrg server -ns opc -set OPC_IGNORE_LICENSE_MGR TRUE
```

The licensing component ignores the license manager when the configuration setting is set to `TRUE`. All nodes displayed in the license report are considered to have the current physical management server as their license manager.

Note: An agent sends license data once a day to the management server set as the primary manager. If a node never had this server as the primary manager, the management server has no license information for the node.

Host Names and IP Addresses

Host Names work within IP networks to identify a managed node. While a node may have many IP addresses, the hostname is used to pinpoint a specific node. The system hostname is the string returned when you use the UNIX `hostname(1)` command.

It is not uncommon for a node to have more than one IP address. If a node becomes a member of another subnet, you may need to change its IP addresses. In this case, the IP address or fully qualified domain name may change.

Note: For HTTPS-based nodes, you can also specify the IP address as dynamic. You can do this by using the `opcnode` command line tool.

In general, on HP-UX and Solaris systems, the IP address and the related hostname are configured in one of the following ways:

- An entry in the file `/etc/hosts`
- Domain Name Service (DNS)
- Network Information Service (NIS on HP-UX, NIS+ on Solaris)

HPOM also configures the hostname and IP address of the management server for the managed node in the management server database.

If you are moving from a non-name-server environment to a name-server environment (DNS or BIND), make sure the name server can access the new IP address.

To change the hostname or IP address of managed nodes, use the `opcchgaddr` command line tool on the management server. For details, see the *opcchgaddr* manual page.

Changing the Hostname or IP Address of the Management Server

To change the hostname or IP address of the management server, follow these steps:

1. Request and install new licenses from the HP Password Delivery Service.

   For more information about HPOM licensing, see the *HPOM Concepts Guide*.
2. Stop all running Java GUIs.
3. Verify that the database is running by using the following command:
   - Oracle database:

     ```
     # ps -ef | grep ora
     ```
   - PostgreSQL database:
     - HP-UX:

       ```
       # /sbin/init.d/ovopsql status
       ```
     - Linux and Sun Solaris:

       ```
       # /etc/init.d/ovopsql status
       ```

   If the database is not running, start it by using the following command:
   - Oracle database:
     - HP-UX:

       ```
       # /sbin/init.d/ovoracle start
       ```
     - Linux and Sun Solaris:

       ```
       # /etc/init.d/ovoracle start
       ```
   - PostgreSQL database:
     - HP-UX:

       ```
       # /sbin/init.d/ovopsql start
       ```
     - Linux and Sun Solaris:

       ```
       # /etc/init.d/ovopsql start
       ```

   For more information about the database, see the *HPOM Installation Guide for the Management Server*.
4. Change the IP address or the node name of the HP Operations management server in the HPOM database by running the following command:

   ```
   /opt/OV/contrib/OpC/opcchgaddr -sync IP <old_IP_addr> <old_FQDN> IP <new_IP_addr> <new_FQDN>
   ```

   For details, see the *opcchgaddr* manual page.
5. Stop all HPOM processes by running the following command:

   ```
   # /opt/OV/bin/ovc -kill
   ```
6. Stop the database by running the following command:

- Oracle database:
  - HP-UX:

    ```
    # /sbin/init.d/ovoracle stop
    ```
  - Linux and Sun Solaris:

    ```
    # /etc/init.d/ovoracle stop
    ```
- PostgreSQL database:
  - HP-UX:

    ```
    # /sbin/init.d/ovopsql stop
    ```
  - Linux and Sun Solaris:

    ```
    # /etc/init.d/ovopsql stop
    ```

7. Update the HP Operations management server configuration by running the following command:

   ```
   # /opt/OV/bin/ovconfchg -ovrg server -ns opc -set OPC_MGMT_SERVER <new_FQDN>
   ```

   Note: The command also updates any other customized settings on the management server, such as `bbc.cb.ports:PORTS`.

8. Update the local agent configuration on the management server as follows:
   a. Specify the new hostname of the management server in the security name space:

      ```
      # /opt/OV/bin/ovconfchg -ns sec.core.auth -set MANAGER <new_FQDN>
      ```
   b. If the certificate server is located on the same system as the management server, update the `CERTIFICATE_SERVER` variable:

      ```
      # /opt/OV/bin/ovconfchg -ns sec.cm.client -set CERTIFICATE_SERVER <new_FQDN>
      ```
9. Update the database files.
   - Oracle database:
     i. In a cluster environment, on each cluster node, replace references to the old hostname with the new hostname. For remote database or cluster nodes also, replace references to the old hostname with the new hostname.

        For example:

        ```
        <Oracle_Install_Dir>/network/admin/listener.ora
        <Oracle_Install_Dir>/network/admin/sqlnet.ora
        <Oracle_Install_Dir>/network/admin/tnsnames.ora
        <Oracle_Install_Dir>/network/admin/tnsnav.ora
        ```

        In these instances, *<Oracle_Install_Dir>* is the directory where you installed Oracle, for example: `/u01/app/oracle/product/11.1.0/db_1`.

     ii. If the `/var/opt/oracle/scls_scr/<old_hostname>` directory exists, rename it to the following location: `/var/opt/oracle/scls_scr/<new_hostname>`.

- PostgreSQL database:

  In a cluster environment, on each cluster node, replace references to the old hostname with the new hostname. For remote database or cluster nodes also, replace references to the old hostname with the new hostname.

  For example:

  ```
  <PostgreSQL_Cluster_Dir>/hpom.conf
  <PostgreSQL_Cluster_Dir>/postgresql.conf
  <PostgreSQL_Cluster_Dir>/pg_hba.conf
  <PostgreSQL_OS_DBA_user_HomeDir>/.pgpass
  /etc/opt/OV/share/conf/ovdbconf
  ```

10. Start the database by running the following command:
    - Oracle database:
      - HP-UX:

        ```
        # /sbin/init.d/ovoracle start
        ```

      - Linux and Sun Solaris:

        ```
        # /etc/init.d/ovoracle start
        ```

    - PostgreSQL database:
      - HP-UX:

        ```
        # /sbin/init.d/ovopsql start
        ```

      - Linux and Sun Solaris:

        ```
        # /etc/init.d/ovopsql start
        ```

11. Start all HPOM processes by running the following command:

    ```
    # /opt/OV/bin/ovc -start
    ```

12. Deploy the modified node configuration to the agent on the management server by running the `opcsw` command locally on the HPOM management server, as follows:

    ```
    # opcsw -get_nodeinfo <new_FQDN>
    ```

    The command writes a temporary file that is read by the distribution agent (for example, when the agent starts or restarts) and creates the appropriate `nodeinfo` file.

    For more information about the `opcsw` command, see the *opcsw(1m)* manual page.

13. To be able to start Java GUI by using the Java Web Start method, update the following line in the `/opt/OV/www/htdocs/ito_op/ito_op_ws.jnlp` file with the new hostname:

    ```
    codebase="http://<new_FQDN>:8081/ITO_OP/">
    ```

14. Reconfigure the HP Operations management server system with the new hostname or IP address:

a. Change the hostname or IP address:

- HP-UX:

  Run the special initialization script `/sbin/set_parms`. For more information about available parameters and options, see the *set_parms(1m)* manual page.

  For details, see the HP-UX System Manager's Guide.

- Linux:

  Run the network configuration tool `system-config-network`. For more information, see the RHEL documentation.

- Sun Solaris:

  Run the `/usr/sbin/sys-unconfig` command. For more details, see the *sys-unconfig(1m)* manual page.

  If you are moving from an environment that does not provide a name resolution service to one that does, make sure the name server has the new hostname or IP address available.

b. Restart the system for your changes to take effect.

15. Update the Administration UI configuration as described in "Changing the Hostname" on page 436.

Reconfiguring the Management Server after a Hostname Change

To reconfigure the management server after changing its hostname or IP address, follow these steps:

1. Stop the HP Operations management server by running the following command:

   ```
   # /opt/OV/bin/OpC/opcsv -stop
   ```

2. Make sure the database is running.

   If the database is not running, start it by using the following command:

   - Oracle database:
     - HP-UX:

       ```
       # /sbin/init.d/ovoracle start
       ```

     - Linux and Sun Solaris:

       ```
       # /etc/init.d/ovoracle start
       ```

   - PostgreSQL database:
     - HP-UX:

       ```
       # /sbin/init.d/ovopsql start
       ```

     - Linux and Sun Solaris:

       ```
       # /etc/init.d/ovopsql start
       ```

For more information about the database, see the *HPOM Installation Guide for the Management Server*.

3. Start the HPOM processes:

   Start the server and agent processes on the HP Operations management server, as follows:

   a. To start the HP Operations management server processes, run the following command on the management server:

      ```
      # /opt/OV/bin/OpC/opcsv -start
      ```

   b. To start the HP Operations agent processes on the management server, run the following command on the management server:

      ```
      # /opt/OV/bin/ovc -start
      ```

      Note: When you restart the agent processes, the agent starts forwarding the messages it buffered while the processes were stopped.

4. Log on to the Java GUI:

   ```
   # /opt/OV/bin/OpC/ito_op
   ```

5. Verify policy assignments to the renamed node.

   Verify that all policies are still assigned to the new node.

6. Redistribute all event correlation policies if you have changed the hostname of the HP Operations management server.

   To redistribute all event correlation policies assigned to the management server, run the `opcragt` command with the `-dist(ribute)` parameter, as follows:

   ```
   # opcragt -dist -force "$MGMTSV"
   ```

   The `$MGMTSV` string is the hostname of the management server.

7. Inform all managed nodes of the new hostname of the HP Operations management server.

   To instruct managed nodes to use the new hostname for the HP Operations management server, perform the following steps on all HTTPS-based managed nodes that are configured in the node bank and which are running an HP Operations agent:

   a. Stop all HP Operations agent processes on the managed nodes:

      ```
      # /opt/OV/bin/ovc -kill
      ```

   b. Specify the new hostname of the management server in the security name space (`sec.core.auth`) by running the following command:

      ```
      # /opt/OV/bin/ovconfchg -ns sec.core.auth -set MANAGER <new_FQDN>
      ```

   c. If the certificate server is located on the same system as the management server (which now has a new hostname), you must also update the `CERTIFICATE_SERVER` variable by running the following command:

```
#/opt/OV/bin/ovconfchg -ns sec.cm.client -set CERTIFICATE_SERVER <new_
FullyQualifiedDomainName>
```

8. Change the primary management server.

   If the modified HP Operations management server is configured as a primary manager for some managed nodes, update those managed nodes by running the following command from the modified HP Operations management server:

   ```
   # /opt/OV/bin/OpC/opcragt -primmgr [-all | [-nodegrp <group>…] <node>…]
   ```

9. Verify and redistribute the policies.

   Verify that the policies are still assigned to the managed nodes, and then redistribute the policies.

10. Update configuration in flexible-management environments, as follows:

    - Hostname and IP address:

      Make sure that your hostname and IP address changes are reflected in all configurations and policies across the entire flexible-management environment.

      To find out how to set up, modify, or distribute the policies in a flexible-management environment, see the *opcmom(4)* manual page.

    - Message forwarding:

      If you have set up message forwarding between HP Operations management servers, modify the hostname and the IP address manually on all management servers that have the changed system in their node bank.

      You must also check the message forwarding policy on the management servers for any occurrence of the old hostname or IP address.

      Modify all files in the following directory:

      ```
      /etc/opt/OV/share/conf/OpC/mgmt_sv/respmgrs/
      ```

      Modify the message forwarding policy on the HP Operations management servers, as needed.

Changing the Hostname or IP Address of an HTTPS Managed Node

Perform the following steps to change the hostname or IP address of an HTTPS-based managed node:

1. Before changing the hostname or IP address of a managed node, consider the following points:

   - Flexible-management environment:

     If you are running HPOM in an environment where multiple management servers are distributed throughout different geographically locations (flexible-management), make sure that you perform all steps described in this procedure on all management server systems that control or monitor the modified node.

- DHCP:

  It is possible to set the IP address of the managed node to dynamic by using the `opcnode` command line interface. This allows you to change the IP address of your HPOM managed node in a safer and a more comfortable way.

- Service Navigator:

  If you are using Service Navigator, check the service configuration files. If the service configuration file contains host names and IP addresses, they may need to be changed before you run `opcservice` again. For more information, see the "Service Configuration File" on page 364.

- Saved filter settings:

  Message browsers allow you to save the filter settings, such as `For the Following Symbols and Objects`. If you change the hostname of a managed node, remember to also change the saved filter to reflect the new hostname so that messages from the node (with the changed name) continue to be displayed after the hostname change.

2. Reconfigure the HPOM managed node system with the new hostname or IP address and restart the system.

   On the managed node, change the hostname or IP address of the system as described in the documentation supplied with the operating system. Then restart the system for your changes to take effect.

3. Change the IP address or the node name of the managed node in the HPOM database by running the following command:

   `/opt/OV/contrib/OpC/opcchgaddr -sync IP <old_IP_addr> <old_FQDN> IP <new_IP_addr> <new_FQDN>`

   For details, see the *opcchgaddr* manual page.

Duplicate IP Addresses for Different Managed Nodes

HPOM enables you to have duplicate IP addresses for different managed nodes. Having duplicate IP addresses for different managed nodes is helpful when you manage an environment with independent subnets that have overlapping IP addresses, which are unique within a network in a private IP address range, but are not unique globally. Because the normal routing does not work for different nodes with the same IP address, make sure that different nodes with the same IP address can be reached through HTTPS proxies.

Note: It is also possible to handle overlapping IP addresses in different networks by using Network Address Translation (NAT). If you already set up NAT in your environment to handle

overlapping IP addresses in different networks, you can safely ignore the procedure described in "Handling Managed Nodes in a Duplicate IP Environment" on page 696. Instead, for more information about NAT, see the *HPOM Firewall Concepts and Configuration Guide*.

By default, HPOM does not allow having duplicate IP addresses for different managed nodes and therefore issues an error.

To enable duplicate IP addresses, run the following command:

```
ovconfchg -ovrg server -ns opc -set OPC_ALLOW_DUPLICATE_IP TRUE
```

`OPC_ALLOW_DUPLICATE_IP` enables you to add two nodes with the same IP address to the node bank.

To receive proxy messages for nodes with overlapping IP addresses (that is, node names matched by an external node pattern but are not in the node bank), set the following additional configuration setting to the command:

```
/opt/OV/bin/ovconfchg -ovrg server -ns opc -set OPC_ALLOW_DUP_IP_PROXY_MSGS TRUE
```

Handling Managed Nodes in a Duplicate IP Environment

When handling managed nodes with duplicate IP addresses, consider that some special steps are required and that certain restrictions apply. Therefore, make sure to carefully follow these steps:

1. **Use HTTP proxies.**

   For networks with overlapping IP addresses, you need a separate HTTP proxy or a chain of proxies. A management server or management servers must have a route to the HTTP proxy. The HTTP proxy must be capable to resolve the managed nodes in that network and have a route to the managed nodes.

   Using HTTP Proxies

   Assume that the service provider `svp.com` handles managed nodes in network `A` for customer A, and managed nodes in network B for customer B. Now assume that both networks use the same IP address range. Proxy PA is used to reach all the nodes in network A and proxy PB is used to reach all the nodes in network B.

   Note that in case of a proxy chain, the first HTTP proxy (the one to be configured on the management server) can also be identical for both subnets. For a better illustration, in this example different proxies are used.

   On the management server, the proxy setting can be the following:

   ```
   ovconfchg -ns bbc.http -set PROXY "PA:8888+(*.a.com);PB:8888+(*.b.com)"
   ```

   On the managed nodes at customer A, the proxy setting can be the following:

   ```
   ovconfchg -ns bbc.http -set PROXY "PA:8888+(*.svp.com)"
   ```

On the managed nodes at customer B, the proxy setting can be the following:

```
ovconfchg -ns bbc.http -set PROXY "PB:8888+(*.svp.com)"
```

2. **The heartbeat polling type must be set to "RPC Only."**

   By using "RPC Only", the management server does not try to ping the managed node that does not work because there is no direct route to the managed node.

3. **Transfer and install the agent software manually.**

   Because there is no direct route to the managed node, you cannot use the server-based installation. You must transfer and install the agent software manually. For details, see "Installing an Agent in a Duplicate IP Environment" on page 697.

4. **Issue and transfer the certificate manually.**

   Make sure to issue and transfer the certificate to the managed node manually.

5. **Use a unique managed node's fully qualified hostname.**

   The managed node's fully qualified hostname must be unique over all subnets.

Installing an Agent in a Duplicate IP Environment

Because there is usually no direct route from the management server to the agent (only indirectly through an HTTP proxy), it is not possible to deploy the agent from the management server. Therefore, you must install the agent manually. However, later patching or upgrading of agents can be done remotely from the management server.

The automatic certificate deployment is not possible. Because you cannot use the HTTPS channel through the proxy for the certificate deployment, you also cannot grant a certificate in the GUI.

To install an agent in a duplicate IP environment, follow these steps:

1. *Optional:* Add the node to the node bank.
2. *Optional:* Create a default profile, for example:

   ```
   /opt/OV/bin/OpC/opcsw -create_inst_info <node_name>
   ```

3. Install the agent software without the configuration on the managed node by running the following command:

   - On the Unix/Linux node:

     ```
     ./oainstall.sh -install -agent -defer_configure
     ```

   - On the Windows node:

     ```
     cscript oainstall.vbs -install -agent -defer_configure
     ```

   For detailed information, see the HP Operations agent documentation.

4. Configure and activate the managed node.

You can choose between the following two types of configuration:

- Default configuration

  To apply the default configuration, run the following commands:

  ```
  cd /opt/OV/bin/OpC/install/
  ./oainstall.sh -c -a -s <management_server_name> -cs <certificate_server_name>
  ```

- Customized configuration

  To apply the customized configuration (created with `opcsw -create_inst_info`), run the following commands:

  ```
  cd /opt/OV/bin/OpC/install
  ./oainstall.sh -c -a -agent_profile <hex_IP_addr>.i
  ```

  For example:

  ```
  cd /opt/OV/bin/OpC/install
  ./oainstall.sh -c -a -agent_profile /tmp/c0a80101.i
  ```

  Note: On Windows, you must run the following command:

  ```
  cscript oainstall.vbs -c -a -agent_profile <hex_ip_addr>.i
  ```

5. Determine the `OvCoreId` of the managed node by running the following command:

   ```
   ovcoreid
   ```

6. Specify the configuration settings to use the proxy for the communication.

   For example:

   ```
   ovconfchg -ns bbc.http -set PROXY "PA:8888+(*.svp.com)"
   ```

7. Issue a certificate on the management server.

   The usage of the `ovcm` command is the following:

   ```
   ovcm -issue -file <file> -name <nodename>
   [-pass <passphrase>] [-coreid <OvCoreId>]
   ```

   For example:

   ```
   ovcm -issue -file /tmp/agent.cert -name agent.a.com -pass pass -coreid
   43d25e12-a57d-7546-1aac-920bab1e6120
   ```

8. Transfer the certificate to the managed node, and then install the certificate on the managed node.

   For example:

   ```
   ovcert -importcert -file /tmp/agent.cert -pass pass
   ```

9. Check if the node is already present in the node bank. If not, add it.
10. Set the correct `OvCoreID` in the database.

For example:

```
/opt/OV/bin/OpC/utils/opcnode -chg_id node_name=agent.a.com id=43d25e12-a57d-7546-1aac-920bab1e6120
```

11. Update the database and start heartbeat polling for the node.

    After the node is connected to the network, on the HP Operations management server, run the following command:

    ```
    /opt/OV/bin/OpC/opcsw -installed <node>
    ```

12. Verify that the HP Operations agent is running on the managed node by typing the following command:

    ```
    /opt/OV/bin/OpC/opcragt -status <node>
    ```

Host Names and IP Addresses in a Cluster Environment

Host names work within IP networks to identify a managed node. Although a node can have many IP addresses, the hostname is used to identify a specific node. The system hostname is the string returned when you use the UNIX `hostname(1)` command.

It is not uncommon for a node in a cluster environment to have more than one IP address. If a node becomes a member of another subnet, you may need to change its IP addresses. In this case, the IP address or fully qualified domain name may change.

Note: For the HTTPS-based nodes, you can also specify the IP address as dynamic. You can do this by using the `opcnode` command line tool.

In general, on HP-UX and Solaris systems, the IP address and the related hostname are configured in one of the following ways:

- `/etc/hosts`
- Domain Name Service (DNS)
- Network Information Service (NIS on HP-UX, NIS+ on Solaris)

HPOM also configures the hostname and IP address of the management server for the managed node in the management server database.

If you are moving from a non-name-server environment to a name-server environment (DNS or BIND), make sure the name server knows about the new IP address.

Changing the Virtual Hostname or IP Address of the Management Server

To change the hostname (or IP address) assigned to the virtual node that is hosting the high-availability resource group (HARG) for the HPOM management server, perform the steps described in the following procedure. Note that, except where otherwise stated, the steps must be performed on the active physical cluster node, where the HP Operations management server resource group (package) is running:

1. Disable monitoring for the HP Operations management server.

   To disable monitoring, enter the following command:

   ```
   # /opt/OV/lbin/ovharg -monitor <om_HARG> disable
   ```

   In this instance, *<om_HARG>* is the name of the high-availability resource group that includes the HPOM management server for which you want to disable monitoring. The default name for the resource group is `ov-server`.
2. Unassign the HPOM management-server policies from the virtual node, whose hostname or IP address you want to change.
3. Stop all HPOM processes on your management server.

   Stop the manager, agent, and Java GUI processes running on the system:

   a. Stop all running Java GUIs.
   b. Stop the HPOM manager processes by entering:

      ```
      # /opt/OV/bin/OpC/opcsv -stop
      ```

   c. Stop the HPOM agents on your management server by entering:

      ```
      # /opt/OV/bin/ovc -kill
      ```

   d. Verify that no HPOM processes are running by entering:

      ```
      # ps -eaf | grep opc
      # ps -eaf | grep ovc
      # ps -eaf | grep coda
      # ps -eaf | grep bbc
      ```

   e. If any HPOM processes are still running, stop them manually by entering the following command:

      ```
      # kill <proc_id>
      ```

      All HPOM agents on HPOM managed nodes start buffering their messages.
4. Make sure the database is running.

   If the database is not running, start it by entering:

- Oracle database:

  ```
  # /sbin/init.d/ovoracle start force
  ```

- PostgreSQL database:

  ```
  # /sbin/init.d/ovopsql start
  ```

For more information about the database, see the *HPOM Installation Guide for the Management Server*.

5. *Oracle only:* Change the HPOM database entry for the IP address or the node name of the virtual cluster node hosting the high-availability resource group for the HP Operations management server by running the following command:

   ```
   /opt/OV/contrib/OpC/opcchgaddr -sync IP <old_IP_addr> <old_FQDN> IP <new_IP_ADDR> <new_FQDN>
   ```

   For details, see the *opcchgaddr* manual page.

6. Shut down the database by running the following command:

   - Oracle database:

     ```
     # /sbin/init.d/ovoracle stop force
     ```

   - PostgreSQL database:

     ```
     # /sbin/init.d/ovopsql stop
     ```

7. Modify the HP Operations management server configuration.

   To change the hostname of the HPOM management server, perform the following steps:

   a. Specify the new hostname of the management server in the security name space:

      ```
      # ovconfchg -ns sec.core.auth -set MANAGER <new_FQDN>
      ```

      In this instance, *<new_FQDN>* is the fully qualified domain name of the virtual cluster node that is now (new) managing the HPOM management-server cluster package (resource group).

   b. Update the HP Operations management server configuration, enter the following:

      ```
      # ovconfchg -ovrg server -ns opc -set OPC_MGMT_SERVER <new_FQDN>
      ```

      In this instance, *<new_FQDN>* is the fully qualified name of the virtual cluster node now managing the HPOM management server cluster package (resource group).

   c. If the certificate server is located on the same system as the management server, update the `CERTIFICATE_SERVER` variable by running the following command:

      ```
      # ovconfchg -ovrg server -ns sec.cm.client -set CERTIFICATE_SERVER <new_FQDN>
      ```

      In this instance, *<new_FQDN>* is the fully qualified name of the virtual cluster node now managing the HPOM management server cluster package (resource group).

   d. Specify the bind address for the server port by running the following command:

```
# ovconfchg -ovrg server -ns bbc.cb -set SERVER_BIND_ADDR <new_IP_addr>
```

In this instance, *<new_IP_addr>* is the IP address of the virtual cluster node now managing the HPOM management server cluster package (resource group).

8. Assign the physical cluster nodes running the HPOM management-server software to the new virtual node that manages the cluster package (resource group) for the HPOM management server:

   Use the `opcnode` command to check if any physical nodes are currently assigned to the virtual node, for example:

   ```
   # opcnode -list_virtual node_name=<new_FQDN>
   ```

   In this instance, *<new_FQDN>* is the fully qualified name of the virtual cluster node now managing the HPOM management server cluster package (resource group).

   To assign physical cluster nodes to a virtual node, use the following command:

   ```
   # opcnode -set_virtual node_name=<new_FQDN> node_list="<PhysicalNode1_
   FQDN><PhysicalNode2_FQDN>" cluster_package="<HARG_Name>"
   ```

<new_FQDN>	Fully qualified name of the virtual cluster node now managing the HPOM management-server cluster package (resource group).
<PhysicalNode#_FQDN>	Fully qualified names of the physical cluster nodes where the HPOM management-server software is installed. Node names in the list are separated by a space character.
<HARG_name>	Name of the HPOM management-server cluster package (resource group).

9. Update the database files:

   - Oracle database:

     On each cluster node, replace references to the old hostname with the new hostname, for example, in the following files:

     ```
     <Oracle_Install_Dir>/network/admin/listener.ora
     <Oracle_Install_Dir>/network/admin/sqlnet.ora
     <Oracle_Install_Dir>/network/admin/tnsnames.ora
     <Oracle_Install_Dir>/network/admin/tnsnav.ora
     ```

     Note that *Oracle_Install_Dir* is the directory where you installed Oracle, for example: `/u01/app/oracle/product/11.1.0/db_1`.

   - PostgreSQL database:

     On each cluster node, replace references to the old hostname with the new hostname, for example, in the following files:

     ```
     <PostgreSQL_Cluster_Dir>/hpom.conf
     <PostgreSQL_Cluster_Dir>/postgresql.conf
     <PostgreSQL_Cluster_Dir>/pg_hba.conf
     ```

```
<PostgreSQL_OS_DBA_user_HomeDir>/.pgpass
/etc/opt/OV/share/conf/ovdbconf
```

10. Start HPOM integrated services, using the following command:

    ```
    # /opt/OV/bin/ovc -start>
    ```

11. Reassign the HPOM management server policies to the virtual node, whose hostname or IP address you have changed.
12. Configure the new high-availability cluster by performing the following steps:
    a. Stop the HPOM-server high-availability resource group by using the `ovharg_config` command with the `-stop` option, as follows:

       ```
       # /opt/OV/bin/ovharg_config <om_HARG> -stop <node_name>
       ```

       In this instance, *<om_HARG>* is the name of the high-availability resource group that includes the HPOM management server for which you want to disable monitoring. The default name for the resource group is `ov-server`.
    b. Change the cluster configuration to use the new IP address.

       For HP Serviceguard:

       Replace the entry `IP[0]=<old_IP_addr>` with `IP[0]=<new_IP_addr>` in the follow file on all cluster nodes.

       ```
       /etc/cmcluster/ov-server/ov-server.cntl
       ```

    c. Start the HPOM-server high-availability resource group as follows:

       ```
       # /opt/OV/bin/ovharg_config <om_HARG> -start <node_name>
       ```

Reconfiguring the Management Server After a Virtual Hostname Change

To reconfigure the management server after changing the name (or IP address) of the virtual node hosting the HPOM resource group (or package) in a cluster environment, perform the following steps:

1. Disable monitoring of the high-availability resource group (HARG).

   To disable HARG monitoring, enter the following command:

   ```
   # /opt/OV/lbin/ovharg -monitor ov-server disable
   ```

2. Stop the HPOM management-server processes.

   To stop the HPOM server processes, enter the following command:

   ```
   # /opt/OV/bin/OpC/opcsv -stop
   ```

3. Make sure the database is running.

   If the database is not running, start it with the following command:

- Oracle database:

  ```
  # /sbin/init.d/ovoracle start
  ```

- PostgreSQL database:

  ```
  # /sbin/init.d/ovopsql start
  ```

For information about managing the database, see the *HPOM Installation Guide for the Management Server*.

4. Start HPOM and all integrated services by using the `opc` command as follows:

   ```
   # /opt/OV/bin/OpC/ovc -start
   ```

5. Enable monitoring of the high-availability resource group (HARG).

   To enable HARG monitoring, enter the following command:

   ```
   # /opt/OV/lbin/ovharg -monitor ov-server enable
   ```

   Note: When you re-enable monitoring for the high-availability resource group, the agent starts forwarding the messages it buffered while the HA resource group was offline.

6. Log on to the Java GUI.

   To start the Java GUI and log on to HPOM, enter the following command:

   ```
   # /opt/OV/bin/OpC/ito_op
   ```

7. Verify HPOM policy assignments.

   Verify that the policies are still assigned to the new node.

8. Reassign and redistribute all event-correlation policies.

   If you have changed the name of the virtual host on which the HPOM management server runs, reassign and redistribute all event-correlation policies assigned to the management server using the `opcragt` command as follows:

   ```
   # opcragt -dist -force "$MGMTSV"
   ```

   The string `$MGMTSV` specifies the name of the host where the HPOM management server is installed.

9. Inform managed nodes about the new (virtual) hostname of the management server.

   To inform managed nodes about a change to the virtual node name of the management server, perform the following steps on HTTPS-based managed nodes that are configured in the node bank and which are running an HPOM agent:

   a. Stop all HPOM agent processes on the managed nodes, enter:

      ```
      # /opt/OV/bin/ovc -kill
      ```

   b. Specify the new (virtual) hostname of the management server in the security name space:

      ```
      #/opt/OV/bin/ovconfchg -ns sec.core.auth -set MANAGER <new_FQDN>
      ```

c. If the certificate server is located on the same system as the management server, update the `CERTIFICATE_SERVER` variable using the `ovconfchg` command as follows:

```
# /opt/OV/bin/ovconfchg -ns sec.cm.client -set CERTIFICATE_SERVER <new_FQDN>
```

d. Restart all HPOM agent processes by entering:

```
# /opt/OV/bin/ovc -start
```

10. Change the primary management server.

    If the modified HP Operations management server is configured as a primary manager for some managed nodes, update those managed nodes by running the following command from the modified HP Operations management server:

```
# /opt/OV/bin/OpC/opcragt -primmgr [-all | [-nodegrp <group>…] <node>…]
```

11. Verify and redistribute the policies.

    Verify that the policies are still assigned to the managed nodes. Then redistribute the policies.

12. Update the configuration files that define flexible-management environments, as follows:

    a. Make sure that your hostname and IP address changes are reflected in all configurations and policies across the entire flexible-management environment. Modify all files in the following directory:

    `/etc/opt/OV/share/conf/OpC/mgmt_sv/respmgrs/`

    To find out how to setup, modify, or distribute the policies in a flexible-management environment, see the *opcmom(4)* manual page.

    b. If you have set up message forwarding between management servers, update any references to the old hostname and IP address on all management servers that include in their node bank the systems whose name or IP address you have changed.

    c. Check the message-forwarding policy on the management servers for occurrences of the old hostname or IP address, for example, in the following file:

    `/etc/opc/OV/share/conf/OpC/mgmt_sv/respmgrs/msgforw`

    Modify the message-forwarding policy on the management servers, if necessary.

    Note: Before setting up flexible-management environment, see the HP Operations Agent documentation for information about security certificates.

13. Change the hostname or IP address of a managed node.

    If you also want to change the hostname or IP address of a managed node, see "Changing the Hostname or IP Address of an HTTPS Managed Node" on page 694.

Improving HPOM Name Resolution

Problems with domain name resolution (DNS) can lead to a reduction in the speed with which HPOM processed messages. To improve HPOM name resolution and message processing speed, check the following configuration details:

1. Make sure reverse DNS lookup is working.
2. Make sure unknown hosts and IP addresses resolve in a reasonable time.
3. If DNS works well, configure your systems to use DNS first and then fallback to `/etc/hosts` in `/etc/nsswitch.conf`:

   ```
   hosts: dns [NOTFOUND=continue] files
   ```

   Note: `opcmsgm` processes messages immediately after the server restart even if the name-resolution service is slow. This is due to the fact that the IP mapping table is created in a separate thread.

   It is also possible to disable the IP mapping table using the `ovconfchg` command as follows:

   ```
   # ovconfchg -ovrg server -ns opc -set OPC_DISABLE_IP_MAPPING_TABLE TRUE
   ```

4. Cache name service results either by setting up the caching DNS server on the HPOM management server or by increasing the size of the HPOM name-service cache.

   If you want to increase the size of the HPOM name-server cache, make sure it is large enough to hold the names of all nodes in the node bank and some additional node names, too. For example:

   ```
   # ovconfchg -ovrg server -ns opc -set OPC_NAMESRV_CACHE_SIZE 10000
   ```

5. Adjust the number of times HPOM tries to resolve hostnames by using the `OPC_NAMESRV_RETRIES` variable (the default value is `1`).HPOM

   Note: first checks the internal cache for the name service results. If there is an entry for this query in the internal cache, the value from the cache is used. If there is no entry for this query in the internal cache, the name service lookup is started and repeated as many times as specified by using the `OPC_NAMESRV_RETRIES` variable.

   For example:

   ```
   # ovconfchg -ovrg server -ns opc -set OPC_NAMESRV_RETRIES 2
   ```

   Note: The default value of `OPC_NAMESRV_RETRIES` is usually adequate for most of the environments. However, if more tries are needed in your environment to resolve hostnames, you can increase the value. Increasing the value leads to the increased time for name resolution. For more information on the server configuration variables, see the *HPOM Server Configuration Variables*.

6. Measure the time it takes to resolve a hostname and generate a warning if the threshold is exceeded (for example, 200 milliseconds) by entering the following:

   ```
   # ovconfchg -ovrg server -ns opc -set OPC_NAMESRV_MAX_TIME 200
   ```

7. Define time-outs for name-resolution functions in DNS that limit the time that a name-resolution call takes to complete, if it encounters problems with name-resolution services.

   Defining time-outs for resolver functions differs according to platform, as follows:

   - HP-UX:

     You can modify the following name-resolution settings on HP-UX:

     - `retrans`: retransmission time-out with the default value being 5000 milliseconds
     - `retry`: number of retries with the default value being 4

     On HP-UX systems, you can set the `retrans` and `retry` options in the following ways:

     - System wide: use the file `/etc/resolv.conf`

       To set the time-out to 1 second and retries to 2, add the following lines to `/etc/resolv.conf`:

       ```
       retrans 1000
       retry 2
       ```

     - For specific processes: use the `RES_RETRY` and `RES_RETRANS` environment variables

       You can use the `ovconfchg` command to set the environment variables `RES_RETRY` and `RES_RETRANS` for `ovcd` (and its children) in the `ctrl.env` name space, for example:

       ```
       # ovconfchg -ns ctrl.env -set RES_RETRY 2 -set RES_RETRANS 1000
       ```

   - Solaris:

     You can modify the following settings on Solaris in the same way as on HP-UX, namely system-wide or for specific processes:

     - `retrans`: retransmission time-out (default = 5 seconds)
     - `retry`: number of retries (default = 4)

     Syntax requirements meant that, on Solaris, you must set `retrans` and `retry` as options in the `resolv.conf` file, as follows:

     ```
     options retrans:1
     options retry:2
     ```

   - Linux:

     You can modify the following settings on Red Hat Linux:

     - `timeout`: The amount of time (in seconds) the name resolver waits for a response from a remote name server before retrying. The default value is 5 seconds. Note that `timeout` on Linux corresponds to `retrans` on HP-UX.

- `attempts` : the number of times the resolver will send a query to its name servers before giving up and returning an error to the calling application. The default value is 2. Note that `attempts` on Linux corresponds to `retry` on HP-UX.

These settings can be modified system wide in `/etc/resolv.conf`. For example, to set the retransmission time-out to 1 second and retries to 2, add the following line to `/etc/resolv.conf`:

```
options timeout:1 attempts:2
```

The options keyword of a system `resolv.conf` file can be amended for specific individual processes by setting the environment variable `RES_OPTIONS` to a space-separated list of resolver options, as illustrated in the following example:

```
export RES_OPTIONS="timeout:1 attempts:2"
```

Using opc.hosts in HPOM Name Resolution

The `opc.hosts` configuration file is an extension to the regular name service on the management server system. The `opc.hosts` file contains IP addresses that cannot or should not be known in the standard name service.

The name-address pairs that are stored in `opc.hosts` are considered in the HPOM name resolution checks. However, the regular name service takes precedence over `opc.hosts` in case of duplicate entries (which should be avoided).

The `opc.hosts` syntax is similar to the syntax of the `/etc/hosts` file and is the following:

<IP_address> <fully_qualified_hostname>

After you create the `opc.hosts` file, save it to the following location on the HP Operations management server:

`/etc/opt/OV/share/conf/OpC/mgmt_sv/`

Note: You do not need to enter any address into `opc.hosts` for the systems with an HP Operations agent installed. Messages from such systems are provided with `OvCoreId`, which is used to select the appropriate node in the HPOM database.

The following is an example of using the `opc.hosts` file:

Mapping SNMP Trap Messages to the Appropriate Node in the HPOM Database

Assume that traps are coming from the system A, which has no HPOM agent installed. The system A has a range of multiple IP addresses from x1 through xn where only x1 is known in the name service. To map all traps from the system A to the appropriate node in the database, add all addresses from x2 through xn to `opc.hosts`.

Achieving Optimal Performance in Large Environments

To achieve optimal performance of HPOM in large environments, set the following:

- Server configuration variables
  - OPC_MAX_DIST_REQS (to minimize the number of managed nodes receiving new configuration data at one time):

    ```
    ovconfchg -ovrg server -ns opc -set OPC_MAX_DIST_REQS 100
    ```

  - OPC_RQS_NUM_WORKERS (to increase the number of ovoareqsdr threads available to communicate with agents):

    ```
    ovconfchg -ovrg server -ns opc -set OPC_RQS_NUM_WORKERS 75
    ```

  - OPC_MSG_BULK_INSERT_RATE (to specify the maximum number of messages that can be added to the database in one block):

    ```
    opccfgchg -ovrg server -ns opc -set OPC_MSG_BULK_INSERT_RATE 1
    ```

  - SocketPoll (set to TRUE when monitoring an environment with more than 1024 nodes):

    ```
    ovconfchg –ovrg server –ns xpl.net –set SocketPoll TRUE
    ```

  - AUTO_CONNECTION_CLOSE_INTERVAL (to define an interval after which an inactive connection from a connection pool is closed):

    ```
    ovconfchg -ns bbc.http.ext.opcbbcdist -set AUTO_CONNECTION_CLOSE_INTERVAL 300
    ovconfchg -ns bbc.http.ext.opcragt -set AUTO_CONNECTION_CLOSE_INTERVAL 60
    ```

  - OPC_DUPL_ANNO_ONLY_IF_CHANGED and OPC_MAX_DUPL_ANNO (to limit the number of annotations for duplicates):

    ```
    ovconfchg -ovrg server -ns opc -set OPC_DUPL_ANNO_ONLY_IF_CHANGED TRUE
    ovconfchg -ovrg server -ns opc -set OPC_MAX_DUPL_ANNO 100
    ```

    Note: The maximum number of duplicate message annotations that you can specify by using the OPC_MAX_DUPL_ANNO server configuration variable is 99999 because of a database limitation. Keep in mind that if you set it to 0 or a value greater than 99999, it is ignored and 99999 is used instead.

  - OPC_HBP_DOUBLE_CHECK and OPC_HBP_DOUBLE_CHECK_DELAY_BUFFER (to avoid unnecessary agent buffering messages):

    ```
    ovconfchg -ovrg server -ns opc -set OPC_HBP_DOUBLE_CHECK=TRUE
    ovconfchg -ovrg server -ns opc -set OPC_HBP_DOUBLE_CHECK_DELAY_BUFFER 90
    ```

- `OPC_TT_MAX_RESP_AGE` (to reduce action response buffering time):

  ```
  ovconfchg -ovrg server -ns opc -set OPC_TT_MAX_RESP_AGE 60
  ```

For detailed information about the variables, see the *HPOM Server Configuration Variables* document and the HP Operations agent documentation.

- Resource limits

  List all the current resource limits by running the `ulimit -a` command and make sure that they correspond to the following values:

  ```
  core file size          (blocks, -c) 0
  data seg size           (kbytes, -d) unlimited
  scheduling priority             (-e) 0
  file size               (blocks, -f) unlimited
  pending signals                 (-i) 257461
  max locked memory       (kbytes, -l) 32
  max memory size         (kbytes, -m) unlimited
  open files                      (-n) 1024
  pipe size            (512 bytes, -p) 8
  POSIX message queues     (bytes, -q) 819200
  real-time priority              (-r) 0
  stack size              (kbytes, -s) 10240
  cpu time               (seconds, -t) unlimited
  max user processes              (-u) 257461
  virtual memory          (kbytes, -v) unlimited
  file locks                      (-x) unlimited
  ```

- Kernel parameters
  - Increase the number of file descriptors to 4096 on the management server as follows:
    - HP-UX management servers

      Make sure the `maxfiles` kernel parameter is set to `4096`.

    - Linux management servers

      Increase the maximum number of open files by using the `limits.conf` file:

      ```
      * soft nofile 4096
      * hard nofile 4096
      ```

    - Sun Solaris management servers

      Follow this procedure:

      A. Verify the hard limit by running the following command:

         ```
         ulimit -n -H
         ```

      B. If the hard limit is less than `4096`, add the following command to `/etc/system`:

         ```
         set rlim_fd_max = 4096
         ```

C. Reboot the system.

D. Set the soft limit in `/etc/profile` or root's `.profile`:

```
ulimit -n 4096
```

- *Linux only:* Make sure that the value of the `kernel.sem` parameter is set to `250 32000 100 256`.

- Oracle tuning parameters:

Tuning Parameter	Default Value	Recommended Value
`memory_target`	`500M`	`512M` (or higher)
`db_files`	`50`	`80`
`db_file_multiblock_read_count`	`16`	`32`
`log_buffer`	`65536`	`1572864`

Note: HPOM requires at least three redo logs with the size of 20M each. However, in large environments, it is highly recommended to have five redo logs with the size of 100M each.

Part VIII: Cluster and High Availability

HP Operations Manager (HPOM) enables you to administer management servers in a cluster environment. The High Availability (HA) manager is a solution that enables you to switch a HA resource group from one virtual cluster node to another.

For detailed information about HPOM in a cluster and HA manager, see:

- "HP Operations Management Servers in a Cluster Environment" on page 713
- "High Availability Manager" on page 732

Chapter 18: HP Operations Management Servers in a Cluster Environment

In this Chapter

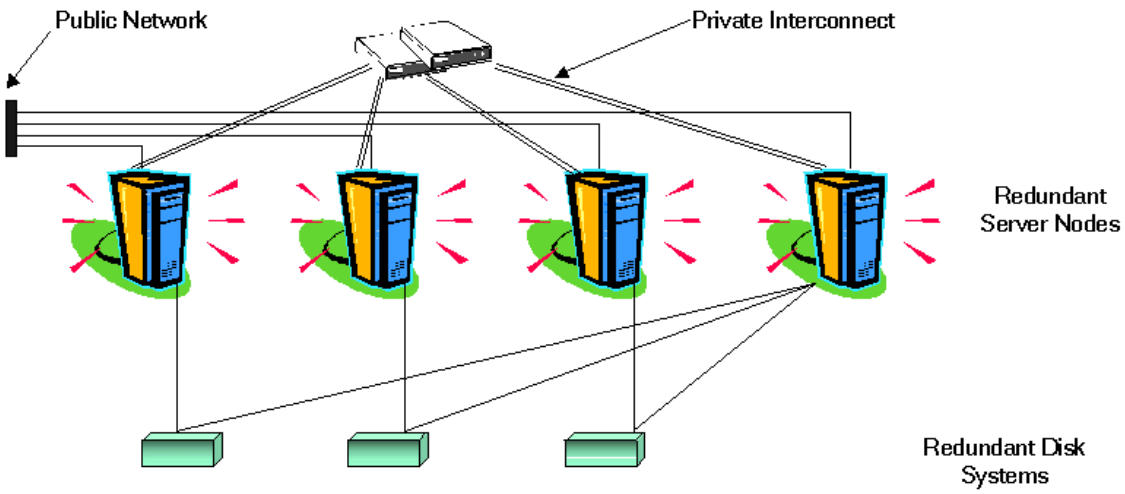
This chapter provides information for system administrators working with HP Operations Manager (HPOM) in a cluster environment. It assumes that you are familiar with the general concepts of HPOM and with high-availability (HA) concepts. The information in this chapter covers the following high-level topics:

For detailed information about installing and configuring the HPOM management server in a high-availability environment, see the *HPOM Installation Guide for the Management Server*.

High-Availability Cluster Environments

Cluster architecture provides a single, globally coherent process and resource management view for the multiple nodes of a cluster. Figure 24 shows an example of a cluster architecture.

Figure 24: Architecture of a High Availability Cluster



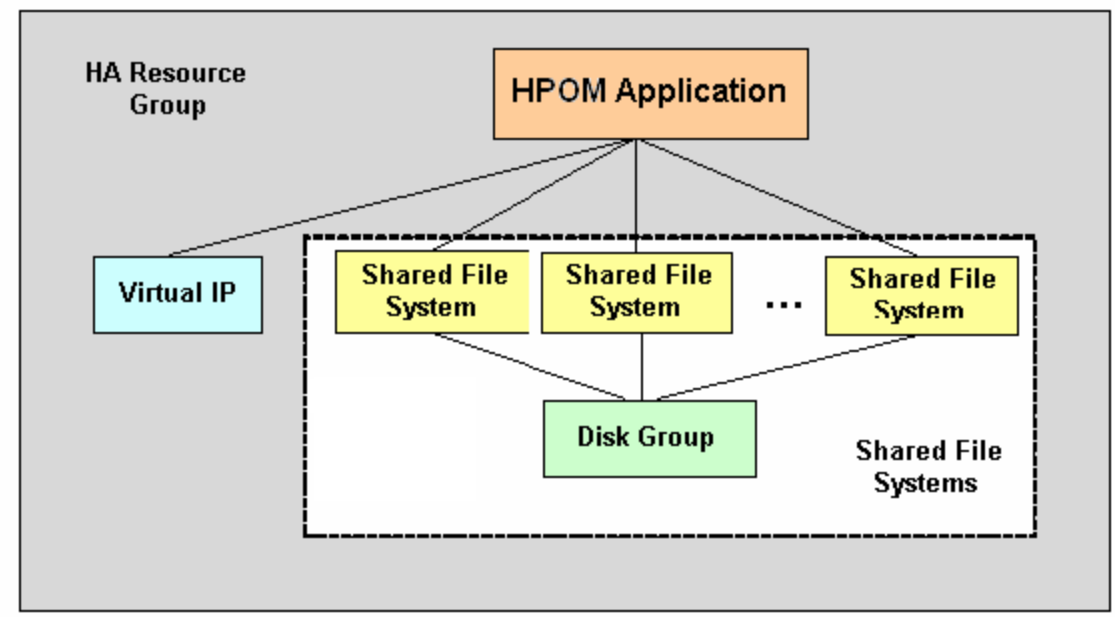
Each node in a cluster is connected to one or more public networks, and to a private interconnect, representing a communication channel used for transmitting data between cluster nodes.

Applications running in a cluster environment are configured as high-availability resource groups. A high-availability resource group (HARG) is a generic term for cluster objects representing highly available applications.

HPOM Management Servers in High-Availability Environments

In modern cluster environments such as HP Serviceguard, VERITAS Cluster, Sun Cluster, Red Hat Cluster, and so on, applications are represented as compounds of resources—simple operations enabling applications to run in a cluster environment. The resources comprise a **Resource Group**, which represents an application running in a cluster environment.

Figure 25: Typical HA Resources Group Layout



The concept of the high-availability resource group is represented differently according to the cluster environment you are talking about. Table 62 indicates how the resource group is referred to in the different high-availability environments.

Table 62: Resource Group in High-Availability Cluster Environments

HA Cluster Environment	Abbreviation	Resource Group Name
HP Serviceguard	HP SG	Package
VERITAS Cluster Server	VCS	Service Group
Sun Cluster	SC	Resource Group
Red Hat Cluster Suite	RH Cluster Suite	Service

Rather than refer to the different, product-specific cluster terms listed in Table 62, this document uses the generic term high-availability resource group (HARG) to designate a set of resources in a cluster environment.

High-Availability-Resource-Group Administration

HPOM provides the `ovharg_config` command to enable you to perform the common tasks required for the administration of HPOM management server running in a high-availability resource group. You can use the `ovharg_config` command to start and the HA resource group and switch the resource group between cluster nodes. The information in this section covers the following topics:

Checking the High-Availability-Resource-Group Status

Before starting, stopping, or switching the high-availability resource group, you can check whether the target node is active:

```
# /opt/OV/bin/OpC/opcsv –startable
```

The `opcsv` command uses the following return codes with the `-startable` parameter:

`0`	Active cluster node is detected.
`1`	Inactive cluster node is detected.

To avoid starting optional processes whose initial configuration is not complete or requires some additional manual steps, use the `opcsv` command to check process availability, as follows:

```
# /opt/OV/bin/OpC/opcsv –available [<process1> <process2> <...>]
```

The `opcsv` command uses the following return codes with the `-available` parameter:

`0`	All specified processes are properly configured, or no processes were specified.
`1`	Not all specified processes are properly configured.

Starting the High-Availability Resource Group

To start the high-availability resource group hosting the HPOM management server, use the `ovharg_config` command with the following parameters:

```
# /opt/OV/bin/ovharg_config <om_HARG> -start <node_name>
```

<om_HARG>	Name of the high-availability resource group hosting the HPOM management server you want to start. The default name for the resource group is `ov-server`.
<node name>	Name of the cluster node on which the high-availability resource group should start.

Note: By default, the resource group name for the HPOM management server cluster is `ov-server`, but you can also choose to specify an alternative name.

The `ovharg_config` command displays the following return codes:

0	HPOM application started successfully.
1	Start operation failed.

Stopping the High-Availability Resource Group

To stop the high-availability resource group hosting the HPOM management-server, use the ovharg_config command with the following parameters:

```
# /opt/OV/bin/ovharg_config <om_HARG> -stop <node_name>
```

<om_HARG>	Name of the high-availability resource group hosting the HPOM management server you want to stop. The default name for the resource group is ov-server.
<node name>	Name of the cluster node on which the high-availability resource group should stop.

The ovharg_config command displays the following return codes:

0	HPOM resource group stopped successfully.
1	Resource-group stop operation failed.

Switching the High-Availability Resource Group

To switch the high-availability resource group from one cluster node to another, use the following command:

```
# /opt/OV/bin/ovharg_config <om_HARG> -switch <node_name>
```

<om_HARG>	Name of the high-availability resource group hosting the HPOM management server you want to switch nodes. The default name for the resource group is ov-server.
<node name>	Name of the cluster node that the high-availability resource group should switch to and start.

The ovharg_config command displays the following return codes:

0	HPOM resource group switched successfully.
1	Resource-group switch operation failed.

Management of the HPOM Management Server in Cluster Environments

The HP Operations management server in a cluster environment is represented as an application that is part of the high-availability resource group, containing resources which perform all necessary operations for starting, stopping and monitoring the application.

HPOM provides the `ovharg` utility to enable you to manually start, stop, and monitor the HP Operations management server when it is running as an application in a cluster environment. For more information about using the `ovharg` utility to help you manage the HPOM management-server resource group in a high-availability environment, see the sections that follow.

Caution: You cannot use the information from this section for managing the HA resource groups. If you try to use the `ovharg` command for starting or stopping the HA resource groups, the operation fails. For instructions on how to manage the HA resource groups, see "High-Availability-Resource-Group Administration" on page 715.

Starting the HPOM Management Server

To start the HP Operations management server manually in a high-availability cluster environment, use the `ovharg` command as follows:

```
# /opt/OV/lbin/ovharg -start <om_HARG>
```

<om_HARG>	Name of the high-availability resource group hosting the HPOM management server you want to start. The default name for the resource group is `ov-server`.

The `ovharg` command displays the following return codes for the `-start` parameter:

`0`	HPOM management server was started successfully.
`1`	Start operation failed.

Stopping the HPOM Management Server

To stop the HP Operations management server manually in a high-availability cluster environment, use the `ovharg` command as follows:

```
# /opt/OV/lbin/ovharg -stop <om_HARG>
```

<om_HARG>	Name of the high-availability resource group hosting the HPOM management server you want to stop. The default name for the resource group is `ov-server`.

The `ovharg` command displays the following return codes for the `-stop` parameter:

0	HPOM management server was stopped successfully.
1	Stop operation failed.

Monitoring the HPOM Management Server

To configure the cluster manager to monitor the HPOM management server in a high-availability cluster environment, use the ovharg command with the -monitor parameter, as follows:

```
# /opt/OV/lbin/ovharg -monitor <om_HARG>
```

<om_HARG>	Name of the high-availability resource group hosting the HPOM management server you want to monitor. The default name for the resource group is ov-server.

The ovharg command displays the following return codes for the -monitor parameter:

0	HPOM management server is running normally
1	HPOM management server is not running, which, if it has not already occurred, leads to a switch of the monitored resource group to another node in the high-availability cluster.

Disabling HPOM Management Server Monitoring

There are situations in which you need the HP Operations management server to be stopped, while all other parts of the high-availability resource group should continue to run. In such situations, you will need to disable monitoring manually.

To manually disable monitoring of the HP Operations management server in a high-availability cluster environment, use the ovharg command with the -monitor parameter and the disable option, as follows:

```
# /opt/OV/lbin/ovharg -monitor <om_HARG> disable
```

<om_HARG>	Name of the high-availability resource group hosting the HPOM management server for which you want to disable monitoring manually. Note that the default name for the resource group is ov-server.

If the monitoring process is disabled, you can stop the HP Operations management server in the knowledge that this will not cause the resource group to be switched to another node in the high-availability cluster. If monitoring is disable, the cluster manager does not detect the event, because the code returned by the monitor command remains 0.

Note: After you have finished the manual HP Operations management server administration, you

must restart the HP Operations management server.

To check whether the HP Operations management server is running normally, use the `opcsv` command as follows:

```
# /opt/OV/bin/OpC/opcsv
```

- If the management server is running, enable monitoring again by using the following command:

  ```
  # /opt/OV/lbin/ovharg -monitor <om_HARG> enable
  ```

<om_HARG>	Name of the high-availability resource group hosting the HPOM management server for which you want to enable monitoring. The default name for the resource group is `ov-server`.

- If the HP Operations management server is not running properly, you have to perform additional manual steps in order to return it to a stable running state.

In a deployment where the HP Operations management server runs in a separate resource group from the database server, you can temporarily disable monitoring of the database high-availability resource group with the following command:

```
# /opt/OV/lbin/ovharg -monitor <DB_HARG> disable
```

In this instance, *<DB_HARG>* is the name of the high-availability resource group hosting the database server for which you want to disable monitoring. The default name for the resource group is `ov-db`.

To enable monitoring of the database high-availability resource group, use the `ovharg` command with the `-monitor` parameter and the `enable` option, as follows:

```
# /opt/OV/lbin/ovharg -monitor <DB_HARG> enable
```

Script-based Database Monitor

If the HP Operations management server and the database server are configured as separate high-availability resource groups, the scripts that monitor the status of the high-availability resource group hosting the HP Operations management server can also be used to monitor the status of the high-availability resource group hosting the database.

If you configure the scripts that monitor the status of the resource group hosting the HP Operations management server to monitor the resource group hosting the database, note the information in the following list, which describes how the management server monitor scripts react to the status of the database high-availability resource group:

- Database high-availability resource group is not running:

  If the HP Operations management server high-availability resource group is started before the database high-availability resource group is up and running, the HP Operations management server

high-availability resource group scripts do not start the HP Operations management server processes.

As soon as the database high-availability resource group is running, the HP Operations management server processes are started and the command returns `0`.

- Database high-availability resource group is stopped:

  If the database high-availability resource group is stopped, switched, or experiences a failover, the HP Operations management server processes are also stopped.

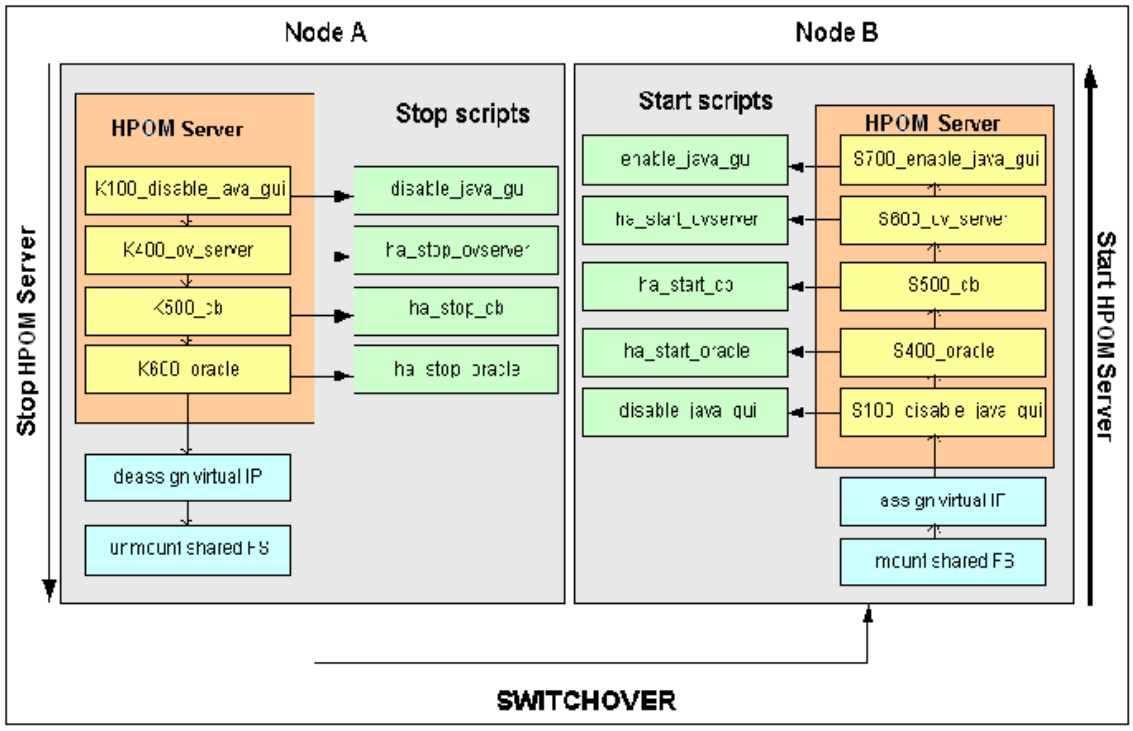
- Database high-availability resource group restarted:

  As soon as the database high-availability resource group is running, the HP Operations management server processes are started and the command returns `0`.

HPOM Switch Over in High-Availability Clusters

The example illustrated in Figure 26 shows the switch-over procedure in a two-node high-availability cluster in which the high-availability resource group `ov-server` is currently active on cluster system `Node A`. The cluster initiates switchover from `Node A` to `Node B`. The resource group `ov-server` is stopped on `Node A` and started on `Node B`. Figure 26 shows the switch-over procedure.

Figure 26: Switchover Procedure



Cluster Switch-Over Process

When a system failure occurs on Node A in the high-availability cluster, the cluster software initiates a switch over of the resource group ov-server by stopping the resource group on Node A and starting it on Node B. The switch over proceeds as follows:

1. On Node A, the cluster-management software performs the following actions:
   a. Cluster manager stops the HP Operations management-server resource group by running the following command:

      /opt/OV/lbin/ovharg -stop *<om_HARG>*

<om_HARG>	Name of the high-availability resource group hosting the HPOM management server you want to stop. The default name for the resource group is ov-server.

      The ovharg script reads all stop links and executes stop scripts in the appropriate sequence.

   b. Cluster manager deassigns the virtual IP from the HPOM management-server resource group and unmounts shared file systems.
2. On Node B, the cluster-management software reforms the following actions:
   a. Cluster manager assigns a virtual IP to the HPOM management-server resource group and mounts shared file systems.
   b. Cluster manager starts the HP Operations management-server resource group by running the following command:

      /opt/OV/lbin/ovharg -start *<om_HARG>*

<om_HARG>	Name of the high-availability resource group hosting the HPOM management server you want to start. The default name for the resource group is ov-server.

      The ovharg script reads all start links and executes start scripts in the appropriate sequence.

The resource group *<om_HARG>* (ov-server) is now active on Node B.

HPOM Troubleshooting in High-Availability Environments

The information in this section helps you to troubleshoot and resolve some of the problems that can occur when HPOM is running in a high-availability environment. In this section, you can find information covering the following topics:

High-Availability Resource Group Does Not Start

If the HPOM resource group cannot be started on any of the nodes in the high-availability cluster, enable tracing to find out why the resource group refuses to start. You can use the information logged in the trace file to help resolve the problem and restart the resource group. In this section you can find instructions to help you perform the following tasks:

Enabling Tracing for the HPOM Resource Group

To enable tracing in the HPOM high-availability resource group, perform the following steps:

1. Make sure that the HPOM high-availability resource group is not running on any cluster node. If the HPOM high-availability resource group is running, stop it with the following command:

   ```
   # /opt/OV/lbin/ovharg_config <om_HARG> -stop <node_name>
   ```

<om_HARG>	Name of the high-availability resource group hosting the HPOM management server you want to stop. The default name for the resource group is `ov-server`.
<node_name>	Name of the high-availability cluster node on which the HPOM resource group that you want to stop is currently running.

2. Enable tracing for the HPOM resource group in the high-availability cluster by using the following command:

   ```
   # /opt/OV/lbin/ovharg -tracing <om_HARG> enable
   ```

3. Restart the HPOM resource group by entering the following command:

   ```
   # /opt/OV/lbin/ovharg_config <om_HARG> -start <node_name>
   ```

   The `ovharg_config` command displays the following return codes:

0	The resource group hosting the HPOM management server started successfully.
1	The resource group hosting the HPOM management server did not start.

If the resource group hosting the HPOM management server does not start, check the output of the trace file, which you can find in the following location on the shared disk on the management server:

/var/opt/OV/hacluster/ov-server/trace.log

If the HPOM management server failed to start, you can try to start it manually by performing the steps described in the section entitled "Starting a Resource Group Manually" on page 724.

Starting a Resource Group Manually

To start the high-availability resource group for the HPOM management server manually, perform the following steps:

1. Mount the shared file systems:
   - File system for the HP Operations server database
   - File system for /etc/opt/OV/share
   - File system for /var/opt/OV/share
   - File system for /var/opt/OV/shared/server
2. Assign the virtual host to the network interface.
3. Start the HPOM resource group using the following command:

   # /opt/OV/lbin/ovharg -start *<om_HARG>*

<om_HARG>	Name of the high-availability resource group hosting the HPOM management server you want to start manually. The default name for the resource group is ov-server.

   The ovharg command displays the following return codes:

0	The resource group hosting the HPOM management server started successfully.
1	The resource group hosting the HPOM management server did not start.

If the resource group hosting the HPOM management server did not start, check the output of the trace file, which you can find in the following location on the shared disk on management server:

/var/opt/OV/hacluster/ov-server/trace.log

If the HPOM management server does not respond to attempts to start it manually using the ovharg command, you can try to start individual components of the resource group using the steps described in the section entitled "Starting Individual Resource Group Components Manually" on page 724.

Starting Individual Resource Group Components Manually

You can manually start individual HP Operations management server components by using the links placed in the following directory:

/var/opt/OV/hacluster/ov-server

When activated, the scripts perform start, stop, and monitor operations for the resource group hosting the HP Operations management server components. The links are given in the following format:

<operation><sequence>_<name>

The following list describes the individual parts of the link:

<operation>	Type of action the link executes, that is, start (S), stop (K), or monitor (M).
<sequence>	Number that indicates the position in the sequence of execution.
<name>	Name of the operation to start.

Note: It is essential to execute links in the correct sequence.

Table 63 lists the links that are available to start up individual components of the HPOM high-availability resource group

Table 63: Resource Group Component Startup

Link Name	Script Location (/opt/OV/bin/OpC/)	Action
S100_disable_java_gui	utils/disable_java_gui	Disables the Java GUI.
S400_oracle or S400_psql	utils/ha/ha_start_oracle or utils/ha/ha_start_psql	Depending on your database type, starts the Oracle or PostgreSQL HARG.
S500_cb	utils/ha/ha_start_cb	Starts the BBC communication broker.
S600_ov_server	utils/ha/ha_start_ovserver	Starts the HP Operations management server HARG.
S700_enable_java_gui	utils/enable_java_gui	Enables the Java GUI.

Table 64 lists the links that are available to stop individual components of the HPOM high-availability resource group

Table 64: Resource Group Component Shutdown

Link Name	Script Location (/opt/OV/bin/OpC/)	Action
K100_disable_java_gui	utils/disable_java_gui	Disables the Java GUI.
K400_ov_server	utils/ha/ha_stop_	Stops the HP Operations management server

Resource Group Component Shutdown, continued

Link Name	Script Location (/opt/OV/bin/OpC/)	Action
	`ovserver`	HARG.
`K500_cb`	`utils/ha/ha_stop_cb`	Stops the BBC communication broker.
`K600_oracle` or `K600_psql`	`utils/ha/ha_stop_oracle` or `utils/ha/ha_stop_psql`	Depending on your database type, stops the Oracle or PostgreSQL HARG.

Table 65 lists the links that are available to enable monitoring for individual components of the HPOM high-availability resource group

Table 65: Resource Group Component Monitoring

Link Name	Script Location (/opt/OV/bin/OpC/utils/)	Action
`M100_oracle` or `M100_psql`	`ha/ha_mon_oracle` or `ha/ha_mon_psql`	Depending on your database type, starts monitoring the Oracle or PostgreSQL HARG.
`M200_cb`	`ha/ha_mon_cb`	Monitors the BBC communication broker in an HA cluster.
`M300_ov_server`	`ha/ha_mon_ovserver`	Starts monitoring the HP Operations management server HARG.

Unplanned Switch Over of the HPOM Management Server HA Resource Group

If specific processes abort and cause an undesired switchover of the high-availability resource group hosting the HP Operations management server, you can temporarily remove the problematic processes from the list of monitored processes.

Disabling Monitoring for Individual Processes

To remove individual processes from the list of processes that you are monitoring in a high-availability environment, perform the following steps:

1. Open the `ha_mon_ovserver` file for editing.

   For more information about the location of the file, see "Resource Group Component Monitoring" on page 726.
2. Disable monitoring for individual processes.

In the list of monitored HPOM management server processes at the end of the file, comment out processes that are causing problems.

Trap Interception in a High-Availability Environment

On the active node in the high-availability cluster, the HPOM event interceptor (`opctrapi`) receives traps from the NNMi postmaster process (`pmd`). After a cluster fail over, `opctrapi` on the now passive cluster node tries to connect to the `pmd` process until the high-availability resource group is switched back again.

Error Handling and Logging in HA Clusters

The scripts that stop, start, and monitor high-availability resource groups (HARG) write information, warnings, and errors to the HA-specific `error.log` file, which you can find in the following location:

`/var/opt/OV/hacluster/<HARG>/error.log`

<HARG>	Name of the high-availability resource group log file you want to read. The default name of the resource group for the HPOM management server is `ov-server`.

The default size of the `trace.log` file for the high-availability resource group is limited. When the maximum file size is reached, `trace.log` is moved to `trace.log.old` and logging information is written into a new `trace.log` file

Setting the Size of the HARG Trace Log

To change the size limit of the `trace.log` file, edit the appropriate parameters in the `settings` file, as follows:

1. Edit the `settings` file for the high-availability resource group hosting the HPOM management server.

   On the HPOM management server, open the `settings` file for editing; you can find the `settings` file in the following location

   `/var/opt/OV/hacluster/<om_HARG>/settings`

<om_HARG>	Name of the high-availability resource group hosting the HPOM management server whose trace-file settings you want to change. The default name for the resource group is `ov-server`.

2. Set the maximum size of the `trace.log` file.

Adding the following line to the `settings` file for the high-availability resource group hosting the HPOM management server whose operations you want to trace:

```
TRACING_FILE_MAX_SIZE=<maximum size in kBytes>
```

For example, to set a maximum size of 7MB, enter the following line:

```
TRACING_FILE_MAX_SIZE=7000
```

HPOM Elements for High-Availability Resource Groups

This section lists and describes the HPOM elements included by default for a high-availability resource group hosting a HPOM management server. The information in this section covers the following areas:

- "HPOM Policies for High-Availability Resource Groups" on page 728
- "HPOM Files for High-Availability Resource Groups" on page 729

HPOM Policies for High-Availability Resource Groups

HPOM provides the following policies and policy groups for high-availability resource groups hosting an instance of the HPOM management server:

- `HA Virtual Management Server`

  The `HA Virtual Management Server` policy group is assigned to the Virtual IP and contains the following policies for the virtual node hosting the HPOM management server:

  - `SNMP 7.01 Traps`
  - `SNMP ECS Traps`

  The trap policy is automatically distributed to all nodes in the high-availability cluster. Since the policy is assigned to the Virtual IP, it is only active on the cluster node where the high-availability resource group (for example, `ov-server`) is currently active.

- `HA Physical Management Server`

  The `HA Physical Management Server` policy group contains the following policies for the physical instance of the HPOM management server:

  - `distrib_mon`
  - `opcmsg (1|3)`
  - `Cron`
  - `disk_util`
  - `mondbfile`

HPOM Files for High-Availability Resource Groups

HPOM and the various cluster-management products it supports stored configuration files, commands, and so on in various directories. The information in this section explains which files are available and where they normally reside.

HP Operations Management Server Files

A selection of files to manage the HPOM management server running in a high-availability environment are located in the following directory on the HP Operations management server:

`/opt/OV/bin/OpC/utils/ha`

The `ha` directory contains the following files:

- `ha_mon_cb`
- `ha_mon_oracle`
- `ha_mon_psql`
- `ha_mon_ovserver`
- `ha_mon_ovserver_3tier`
- `ha_start_cb`
- `ha_start_oracle`
- `ha_start_psql`
- `ha_start_ovserver`
- `ha_start_rg`
- `ha_stop_cb`
- `ha_stop_oracle`
- `ha_stop_psql`
- `ha_stop_ovserver`
- `ha_timeout`

For more information about what the individual commands do, see the various tables in "Starting Individual Resource Group Components Manually" on page 724.

High-Availability Commands

HP Operations Manager provides the following commands to configure and manage an HPOM management server running in a cluster environment:

- `/opt/OV/lbin/ovharg`

  For more information, run the `ovharg` command with the `-help` option.
- `/opt/OV/bin/ovharg_config`

  For more information, run the `ovharg_config` command with the `-help` option.

Product-Specific High-Availability Files

HPOM provides configuration files that you can you use to set up, manage, and monitor HPOM in a cluster environment. The files available and their names differ according to platform and product, as follows:

- HP Serviceguard Files:

  HP Serviceguard specific files are located in the following directory:

  `/opt/OV/lbin/clusterconfig/mcsg`

  The `mcsg` directory contains the following files:
  - `ov_rg.cntl`
  - `ov_rg.conf`
  - `ov_rg.mon`
- Sun Cluster Files:

  You can find Sun Cluster files in the following directory:

  `/opt/OV/lbin/clusterconfig/sc3`

  The `sc3` directory contains the following files:
  - `monitor_start`
  - `monitor_stop`
  - `start`
  - `stop`
  - `probe`
  - `gettime`
  - `HP.OVApplication`

  Additional Sun Cluster files are located in the following directory:

  `/opt/OV/lbin/clusterconfig/sc3/OVApplication`

  The `OVApplication` directory contains the following files:

- monitor
- online
- offline

Chapter 19: High Availability Manager

In This Chapter

This chapter describes the High Availability Manager (HA Manager). The HA Manager enables you to switch a high availability resource group (HARG) between the nodes that make up a virtual cluster.

The following high availability and cluster terms are used in this chapter:

Virtual Cluster	Virtual clusters consist of independent nodes grouped together by the HA Manager, which runs on each node. Virtual clusters also include a HARG (that is, a high availability application in a virtual cluster). Applications that are integrated into a virtual cluster become high availability applications when they are active on one node in the cluster in a particular period of time. If there is a problem with the node on which the application runs, the application activity is moved to another node, which results in the application running smoothly at all times.
HARG	A high availability resource group that represents a resource defined in the “cluster world” that can be linked to an application instance. The HARG runs on a cluster and can be switched from one cluster node to another. A cluster package is usually also linked to an element from the “networking world” known as a virtual node.
Virtual Node	A virtual node is the network representation of an application package running on an HA cluster. A virtual node typically has a hostname and an IP address, it is known to the name resolution, and it can be addressed like an ordinary system.
Virtual IP Address	A virtual IP address is typically part of a HARG and can be switched from one cluster node to another cluster node.
Reference Node	A reference node, which does not belong to the HA Manager environment, is used primarily for checking the connection between the nodes. If a local node fails to make a connection to another cluster node or if the connection is lost, the local node tries to access the reference node (for example, a local DNS server) to pinpoint on which node a problem occurred. The reference node check is used to avoid the “split-brain syndrome” (that is, to find out if the network is down or if the network is up and running, but the other cluster node is down). A “split brain” means that both cluster nodes believe that the other one is down and each of

	them takes over the control, but the real problem lies in the network connection between the nodes and therefore they cannot communicate to each other.

The information in this chapter covers the following topics:

HA Manager and HARG Concepts

High availability is a general term used to characterize environments that represent business-critical systems protected against downtime through different redundant resources. The HA Manager is a light-weight solution that allows the configuration of an automatic failover of the virtual IP address in a server pooling setup in a similar way as in a regular failover cluster. The HA Manager is not an additional cluster software such as the HP Serviceguard or the Red Hat Cluster Suite, but it is an alternative. It represents a cluster without special hardware with redundancy and a shared disk. Therefore, the HA Manager feature enables you to do the following:

- Switch an IP address from one node to another node within a server pooling environment. In this case, no hardware cluster is needed and HP Operations agents and Java GUI instances can communicate using that high availability virtual IP address.

  For details, see the HP Operations High Availability Through Server Pooling document.
- Control other resources besides virtual IP addresses and make them high available.

For more information about high availability, see the *HPOM Concepts Guide*.

Before you start using the HA Manager, you should be familiar with the following concepts:

- **HA Manager communication**

The HA Managers that run on different nodes use a file transfer for communication because it represents a simple, reliable, and standardized way of exchanging data between the systems or the nodes.

The following two directories are used for the file transfer on each node:

- `/var/opt/OV/hamanager/input`
  This directory contains the files that the local HA Manager receives from other nodes. The HA Manager automatically reads the received files and deletes them from this directory after reading them.

- `/var/opt/OV/hamanager/output`
  This directory contains the files that the local HA Manager sends to other nodes. The HA Manager deletes the files from this directory after sending them.

- **HARG status and status synchronization or update**

  The statuses of all nodes are stored in the `/var/opt/OV/hamanager/status` directory. For each node, there is a subdirectory that contains the statuses of all HARGs:

  `/var/opt/OV/hamanager/status/<nodename>`

  The status of each HARG is stored in the `harg_status.<HARG_name>` file.

  The local HA Manager updates the statuses of HARGs for a local node, whereas the remote HA Manager sends the statuses of HARGs from the remote nodes.

- **HA Manager alive check**

  For each node, an alive check is performed by checking the time of the last status update from a selected node. The alive check ensures that a virtual cluster runs smoothly by taking an immediate and adequate action in case a problem occurs on a certain node. A local node or a remote node is detected as `FAULTED` when the following happens:

  - A local node is detected as `FAULTED` when a reference node is down and there is no connection to other nodes (that is, there is no status update from other nodes for some time).

  - A remote node is detected as `FAULTED` when there is no connection to a selected node (that is, there is no status update from the selected node for some time) and when a local node can access a reference node, or when there is no ping reply from the selected node.

- **HARG online, offline, and monitor procedures**

  When a HARG online, offline, or monitor procedure is performed, the HA Manager runs one of the following commands on a selected node:

  `/opt/OV/lbin/ovharg -start <HARG_name>`
  `/opt/OV/lbin/ovharg -stop <HARG_name>`
  `/opt/OV/lbin/ovharg -monitor <HARG_name>`

When starting, stopping, or monitoring the HARG, the ovharg tool performs all the start (S), stop (K), or monitor (M) operations as specified by a sequence number.

The return code is 0 when the action is performed successfully, otherwise the return code is 1.

The general return code is a collection of individual return codes. When all return codes are 0, the general return code is also 0. However, if there is at least one return code that is 1, the general return code is also 1.

Note: HARG monitoring is performed only on the node on which the HARG is ONLINE.

- **HARG autostart**

  A HARG is started automatically on a local node that is ONLINE in the following cases:

  - The HARG is OFFLINE and the local node is the primary node.
  - The HARG is OFFLINE and the primary node is FAULTED.
  - The HARG is FAULTED on some other node, but OFFLINE on the local node.
  - The HARG is ONLINE on a FAULTED node and either the local node is the primary node or the primary node is FAULTED.

  Keep in mind the following specifics:

  - When sending the local status to the remote HA Manager fails, the remote node is not marked as FAULTED after one failure, but after a specified number of failures (the default value is 3). To specify after how many failures the remote node should be marked as FAULTED, set the MAX_COMM_PROBLEMS variable in the HA Manager configuration file to a desired value.
  - The node alive timeout (that is, time during which the remote node status must be updated) can be set by using the NODE_ALIVE_TIMEOUT variable in the HA Manager configuration file (the default value is 60 seconds). If the node status is not updated in specified time, the node becomes FAULTED.
  - The node sends its local status every 15 seconds (the default value). You can set another value by using the MAX_SEND_LOCAL_STATUS_TIME variable in the HA Manager configuration file.
  - The HARG that is FAULTED on a local node can be automatically cleared only if it is ONLINE on some other node. This behavior is disabled by default. To enable it, set the HARG_AUTOCLEAN_FAULTED_TIME variable in the HA Manager configuration file to a number that is greater than zero. This number represents the number of seconds that elapses from the moment the HARG becomes FAULTED until the moment the autoclean is performed.
  - The HA Manager configuration is automatically reloaded when it is modified.

- **HARG autostop**

  A HARG is stopped automatically on a local node if the HARG is `ONLINE` on both the local node that is not the primary node and the primary node. This is because the HARG cannot run on more than one node at the same time.

- **Disabled HARG**

  When a HARG is disabled, the HA Manager cannot control the HARG. Therefore, HARG online, offline, and monitor procedures are disabled.

HA Manager Tool

When switching a HARG between the nodes that make up a virtual cluster, use the `opchamgr` tool that you can find at the following location:

`/opt/OV/bin/OpC/`

The syntax of the `opchamgr` tool is as follows:

```
opchamgr -daemon
         -kill
         -monitor <harg>
         -init
         -status
         -node list
         -node join <active_cluster_node>
         -node unregister <nodename>
         -harg list|add|delete|disable|enable <harg>
         -harg primary <harg> <node>
         -online <harg> [<nodename>]
         -offline <harg>
         -clear <harg> [<nodename>]
         -update
         -ping [<nodename>]
         -trace [enable|disable]
         -h|-\?|-help
```

You can use the following options with the `opchamgr` tool:

`-daemon`	Monitors the HA Manager daemon and starts it if it does not run.
`-kill`	Stops the HA Manager on a local node.
`-monitor <harg>`	Performs HARG status monitoring.
`-init`	Initializes a cluster on a local node.
`-status`	Shows the HA Manager status, nodes, and HARGs.

`-node list`	Lists virtual cluster nodes.
`-node join <active_cluster_node>`	Joins a local node to a virtual cluster that already runs on an active cluster node.
`-node unregister <nodename>`	Unregisters a node from a cluster.
`-harg list|add|delete|disable|enable <harg>`	Lists, adds, deletes, disables, or enables a HARG.
`-harg primary <harg> <node>`	Sets a primary node for a HARG.
`-online <harg> [<nodename>]`	Makes a HARG online on a selected node.
`-offline <harg>`	Makes a HARG offline in a virtual cluster.
`-clear <harg> [<nodename>]`	Clears the `FAULTED` status of a HARG in general or on a selected node.
`-update`	Distributes local configuration to all nodes.
`-ping [<nodename>]`	Checks communication with a cluster node.
`-trace [enable|disable]`	Enables or disables tracing.
`-h|-\?|-help`	Shows the usage.

Caution: It is highly recommended that you consistently use either short hostnames or long hostnames for node names, and avoid combining them.

In addition, when specifying a node name in the `opchamgr` command, make sure to use either the long hostname or the short hostname depending on the form defined for that particular node.

Setting up an HA Manager Environment

To set up an HA Manager environment, follow these steps:

1. On each node, create the following directories:

   ```
   mkdir -p /etc/opt/OV/hamanager
   mkdir -p /var/opt/OV/hamanager/input
   mkdir -p /var/opt/OV/hamanager/output
   ```

2. After you create the directories, create the following HA Manager configuration file on each node:

   ```
   /etc/opt/OV/hamanager/hamanager.conf
   ```

3. Set the following variables in the hamanager.conf file:

   ```
   HAMGR_COMMUNICATION_TYPE=<selected_communication_type>
   REFERENCE_NODE=<reference_node>
   ```

   For example:

   ```
   HAMGR_COMMUNICATION_TYPE=SSH
   REFERENCE_NODE=dnsnode
   ```

   In this instance, HAMGR_COMMUNICATION_TYPE defines a communication type and REFERENCE_NODE is used for testing communication.

   Caution: Make sure that you set up a passwordless connection.

   You can choose among the following communication types:

   - SSH

     Basic SSH communication channel for which you must use the following command:

     ```
     scp <filename> <target_node>:<target_dir>
     ```

   - REMSH

     Basic REMSH communication channel for which you must use the following command:

     ```
     rcp <filename> <target_node>:<target_dir>
     ```

   - CUSTOM_TOOL

     When HAMGR_COMMUNICATION_TYPE is set to CUSTOM_TOOL, an additional variable must be set, CUSTOM_TRANSFER_TOOL. This variable must contain the full path to the tool performing a file transfer from a local node to a target node (that is, from /var/opt/OV/hamanager/output to /var/opt/OV/hamanager/input).

     The tool usage is as follows:

     ```
     <tool> <target_node> <file_to_send>
     ```

     For example, if the CUSTOM_TRANSFER_TOOL is set to /opt/OV/bin/OpC/utils/hamgr_transfer, the tool usage is as follows:

     ```
     /opt/OV/bin/OpC/utils/hamgr_transfer nodeA
     /var/opt/OV/hamanager/output/test.file
     ```

   - CUSTOM_COMMAND

     When HAMGR_COMMUNICATION_TYPE is set to CUSTOM_COMMAND, an additional variable must be set, CUSTOM_TRANSFER_COMMAND. This variable must contain the full command for transferring a selected file from the /var/opt/OV/hamanager/output directory on a local node to the /var/opt/OV/hamanager/input directory on a target node.

     For example:

```
scp /var/opt/OV/hamanager/output/${FILE} ${NODE}:/var/opt/OV/hamanager/input
```

In this instance, `${FILE}` and `${NODE}` are replaced with a filename and a nodename.

> **Note:** If a non-root user is used for a file transfer, set permissions of the `/var/opt/OV/hamanager/input` directory to `777`.

To check if the communication channel works, on each node, perform the following steps (assuming that the file is `commCheck.file` and the target node is `nodeB`):

i. Create the following file:

```
/var/opt/OV/hamanager/output/commCheck.file
```

ii. Replace `$FILE` with `commCheck.file` and `$NODE` with `nodeB`.

For example, to do this, run the following command:

```
scp /var/opt/OV/hamanager/output/commCheck.file
nodeB:/var/opt/OV/hamanager/input
```

iii. On `nodeB`, the following file should exist:

```
/var/opt/OV/hamanager/input/commCheck.file
```

iv. Delete `commCheck.file` on both nodes.

4. *Optional:* Customize ping retries (that is, override the default values) by adding the following variables to the `hamanager.conf` file:
   - `PING_RETRIES`: Number of ping retries (the default value is `3`).
   - `PING_RETRY_DELAY`: Time in seconds before retrying ping (the default value is `5`).
5. Start the HA Manager daemon on both nodes by running the following command:

   ```
   /opt/OV/bin/OpC/opchamgr -daemon
   ```
6. Add the `opchamgr -daemon` command to `crontab` to enable regular HA Manager monitoring and an automatic restart if the HA Manager is stopped.

   For example, for the `opchamgr -daemon` command to be run every hour, add the following line to `crontab`:

   ```
   0 * * * * /bin/sh /opt/OV/bin/OpC/opchamgr -daemon
   ```
7. Initialize a cluster on one node by running the following command:

   ```
   /opt/OV/bin/OpC/opchamgr -init
   ```
8. After a few moments, check the status on the local node by typing the following:

   ```
   /opt/OV/bin/OpC/opchamgr -status
   ```

   The status of the local node should be `ONLINE` and no HARGs are configured at this point.
9. On the second node, run the following command to join this node to the cluster:

   ```
   /opt/OV/bin/OpC/opchamgr -node join <first_active_cluster_node>
   ```

10. After a few moments, check the status on both nodes by typing the following:

    `/opt/OV/bin/OpC/opchamgr -status`

    Both nodes should be listed and their statuses should be `ONLINE`. No HARGs are configured at this point.

Configuring the HP Operations Management Server HARG in a Server Pooling Environment

In a server pooling environment, HP Operations management servers are configured identically and the role of the primary manager is assigned to a virtual interface. Managed nodes send their messages not to a physical server but to its virtual interface.

Note: The procedure described in this section is an example of configuring the HP Operations management server HARG in a server pooling environment.

For detailed information about server pooling, see the *HP Operations High Availability Through Server Pooling* document.

To configure the HP Operations management server HARG in a server pooling environment, first define a HARG name (for example, `hpom-server`), and then follow these steps:

1. *Only if you plan to use the Health Check component:* On all HA Manager nodes, enable the HC component by running the following command:

   `/opt/OV/bin/OpC/utils/hc/opchc.sh -enable [SM]`
2. On each node, create HARG directories and configuration files. To do this, follow these steps:
   a. Create the `/var/opt/OV/hacluster/<HARG_name>` directory by running the following command:

      `mkdir -p /var/opt/OV/hacluster/<HARG_name>`
   b. Change to this newly created directory by running the following command:

      `cd /var/opt/OV/hacluster/<HARG_name>`

      Caution: Before you continue with the next step, make sure that you are familiar with the tools described in "Resources" on page 743.
   c. In the `/var/opt/OV/hacluster/<HARG_name>` directory, create the following links:

      ```
      ln -s /opt/OV/bin/OpC/utils/ha/ha_virtual_ip S100_virtual_ip
      ln -s /opt/OV/bin/OpC/utils/ha/ha_virtual_ip K200_virtual_ip
      ln -s /opt/OV/bin/OpC/utils/ha/ha_virtual_ip M100_virtual_ip
      ln -s /opt/OV/bin/OpC/utils/ha/ha_mon_ovserver M200_server
      ```

      Make sure to create additional links in the following cases:

- In an IPv6 environment:

  ln -s /opt/OV/bin/OpC/utils/ha/ha_virtual_ip S090_virtual_ipv6

  ln -s /opt/OV/bin/OpC/utils/ha/ha_virtual_ip K200_virtual_ipv6

  ln -s /opt/OV/bin/OpC/utils/ha/ha_virtual_ip M100_virtual_ipv6
- If the Health Check component is used:

  ln -s /opt/OV/bin/OpC/utils/ha/ha_HC S800_HC

  ln -s /opt/OV/bin/OpC/utils/ha/ha_HC K050_HC

  Note: When the HC component is enabled on an inactive node, no HC error messages are reported from this node after the HARG is stopped on it for the first time. To avoid HC error messages before stopping the HARG on such a node for the first time, run the following commands:

  ```
  /opt/OV/bin/ovc -stop opchcd
  /opt/OV/bin/ovcreg -del opchcd
  ```

d. *Outbound-only communication only:* In the `/var/opt/OV/hacluster/<HARG_name>` directory, create the following links:

   ln -s /opt/OV/bin/OpC/utils/ha/ha_start_cb S200_ovrg

   ln -s /opt/OV/bin/OpC/utils/ha/ha_stop_cb K100_ovrg

   ln -s /opt/OV/bin/OpC/utils/ha/ha_mon_cb M300_ovrg

e. Set the virtual IP resource configuration in the following file:

   `/var/opt/OV/hacluster/<HARG_name>/resource_virtual_ip.conf`

   The following variables must be set:

   ```
   ADDRESS=<virtual_IP_address>
   DEVICE=<device_to_which_virtual_IP_is_attached>
   NETMASK=<netmask_address>
   ```

   For example:

   ADDRESS=192.168.1.100
   DEVICE=e1000g0:1
   NETMASK=255.255.0.0

   The following variable is optional:

   `OPTIONS=<custom_options_to_ifconfig_command_when_activating_IP>`

   IPv6 environment only: Create the following additional configuration file:

   ```
   /var/opt/OV/hacluster/<HARG_name>/
   resource_virtual_ipv6.conf
   ```

   The following variables must be set in `/var/opt/OV/hacluster/<HARG_name>/resource_virtual_ipv6.conf`:

```
ADDRESS=<virtual_IPv6_address>
DEVICE=<device_to_which_virtual_IPv6_address_is_attached>
NETMASK=<netmask_IPv6_address_suffix>
```

For example:

```
ADDRESS=fec0::250:56ff:fea8:4966
DEVICE=lan0:1
NETMASK=64
```

For details, see "Resources" on page 743.

f. *Outbound-only communication only:* Set the OVRG resource configuration in the following file:

```
/var/opt/OV/hacluster/<HARG_name>/resource_ovrg.conf
```

The following variable must be set:

```
OVRG=<OV_resource_group_name_used_for_outbound-only_configuration>
```

For example:

```
OVRG=virt
```

For details, see "Resources" on page 743.

3. Check virtual IP activation or deactivation.

At this point, the HARG is not under the control of the HA Manager, so you can only check if the `start`, `stop`, and `monitor` commands work. To do this, select a node for the check, make sure that the virtual IP and the OVRG are not active, and then follow these steps:

a. Activate the HARG:

```
/opt/OV/lbin/ovharg -start <HARG_name>
```

The return code should be 0.

b. Check the HARG status:

```
/opt/OV/lbin/ovharg -monitor <HARG_name>
```

The return code should be 0.

The virtual IP should be attached to the network interface. To check this, run the following command:

```
ifconfig -a
```

Outbound-only communication only: The OVRG should be active. To check this, run the following command:

```
/opt/OV/bin/ovbbccb -ovrg `hostname` | grep <ovrg_name>
```

c. Deactivate the HARG:

```
/opt/OV/lbin/ovharg -stop <HARG_name>
```

The return code should be 0.

d. Check the HARG status:

```
/opt/OV/lbin/ovharg -monitor <HARG_name>
```

The return code should be 1.

The virtual IP should not be attached to the network interface. To check this, run the following command:

```
ifconfig -a
```

Outbound-only communication only: The OVRG should not be active. To check this, run the following command:

```
/opt/OV/bin/ovbbccb -ovrg `hostname` | grep <ovrg_name>
```

After you configure the HP Operations management server HARG in a server pooling environment, put the HARG under the HA Manager control. For details, see "Putting a HARG under the HA Manager Control" on page 745.

Resources

You can use the following predefined tools for managing the HP Operations management server, the Oracle database, the virtual IP, and the OVRG that can be used within the HARGs:

- HP Operations management server:
  - For monitoring the HP Operations management server:

    ```
    /opt/OV/bin/OpC/utils/ha/ha_mon_ovserver
    ```

  - For checking the connection between the HP Operations management server and the database server:

    ```
    /opt/OV/bin/OpC/utils/ha_mon_dbconn
    ```

  - For starting the HP Operations management server:

    ```
    /opt/OV/bin/OpC/utils/ha/ha_start_ovserver
    ```

  - For stopping the HP Operations management server:

    ```
    /opt/OV/bin/OpC/utils/ha/ha_stop_ovserver
    ```

- Database:
  - Oracle:
    - For monitoring the Oracle database:

      ```
      /opt/OV/bin/OpC/utils/ha/ha_mon_oracle
      ```

    - For starting the Oracle database:

      ```
      /opt/OV/bin/OpC/utils/ha/ha_start_oracle
      ```

    - For stopping the Oracle database:

      ```
      /opt/OV/bin/OpC/utils/ha/ha_stop_oracle
      ```

- PostgreSQL:
  - For monitoring the PostgreSQL database:

    `/opt/OV/bin/OpC/utils/ha/ha_mon_psql`
  - For starting the PostgreSQL database:

    `/opt/OV/bin/OpC/utils/ha/ha_start_psql`
  - For stopping the PostgreSQL database:

    `/opt/OV/bin/OpC/utils/ha/ha_stop_psql`

- OVRG:
  - For monitoring the OVRG within the Communication Broker:

    `/opt/OV/bin/OpC/utils/ha/ha_mon_cb`
  - For starting the OVRG within the Communication Broker:

    `/opt/OV/bin/OpC/utils/ha/ha_start_cb`
  - For stopping the OVRG within the Communication Broker:

    `/opt/OV/bin/OpC/utils/ha/ha_stop_cb`

- Virtual IP:

  For monitoring, starting, and stopping the virtual IP:

  `/opt/OV/bin/OpC/utils/ha/ha_virtual_ip`

General Purpose Resource

Besides the predefined tools, you can also use a general purpose resource that represents a general resource with real tools and their usage specified in the resource configuration file.

To use the general resource, the following links must be created:

`ln -s /opt/OV/bin/OpC/utils/ha/ha_resource K<sequence>_<name>`

`ln -s /opt/OV/bin/OpC/utils/ha/ha_resource S<sequence>_<name>`

`ln -s /opt/OV/bin/OpC/utils/ha/ha_resource M<sequence>_<name>`

The tool naming convention in these links is predefined and is as follows:

<operation><sequence>_<name>

In these instances, `K`, `S`, or `M` (that is, *<operation>*) represents the type of action the link executes (stop (`K`), start (`S`), or monitor (`M`)), *<sequence>* is the number that indicates the position in the sequence of execution, and *<name>* is the name of the operation to start.

The general resource configuration is set in the following file:

`/var/opt/OV/hacluster/<HARG_name>/resource_<name>.conf`

In this instance, *<name>* is the same as *<name>* used in *<operation><sequence>_<name>*.

The following variables must be set in this file:

```
START_COMMAND=<command_for_starting_this_resource>
STOP_COMMAND=<command_for_stopping_this_resource>
MONITOR_COMMAND=<command_for_monitoring_this_resource>
```

The commands defined in this configuration file return 0 when the action is performed successfully and 1 when the action fails.

An example of a general purpose resource is a resource for managing a web server. If you use this resource, create the appropriate links to the `ha_resource` tool and set the following variables in the `resource_web.conf` file:

```
MONITOR_COMMAND=<command_to_get_web_server_status>
START_COMMAND=<command_to_start_web_server>
STOP_COMMAND=<command_to_stop_web_server>
```

Putting a HARG under the HA Manager Control

To put a HARG under the HA Manager control, follow these steps:

1. On one of the nodes, add the HARG to the HA Manager by running the following command:

   ```
   /opt/OV/bin/OpC/opchamgr -harg add <HARG_name>
   ```

2. Check the HA Manager status on both nodes by typing the following:

   ```
   /opt/OV/bin/OpC/opchamgr -status
   ```

   In the status output, the HARG is listed and its status is `OFFLINE` on both nodes. There is no primary nor active node.

3. Set a primary node for the HARG (that is, a node on which the HARG is `ONLINE` by default) by running the following command:

   ```
   /opt/OV/bin/OpC/opchamgr -harg primary <HARG_name> <node>
   ```

   Note: When the primary node is set, the HARG is automatically started on the selected node. The autostart is performed within one minute.

4. Check the status of the HA Manager by running the following command:

   ```
   /opt/OV/bin/OpC/opchamgr -status
   ```

   The status output shows that the primary node is set. After a few moments, the HARG becomes `ONLINE` on the primary node.

When the HARG is under the control of the HA Manager, you can use the `opchamgr` tool to perform a HARG switchover. For details, see "Performing a HARG Switchover or Failover" on page 746.

Performing a HARG Switchover or Failover

Once a HARG is under the control of the HA Manager, use the `opchamgr` tool to perform a HARG switchover or failover. The difference between the switchover and the failover is that the switchover represents a controlled switch of a cluster package from one cluster node to another (for example, due to load balancing), whereas the failover represents an unplanned switch of a cluster package from one cluster node to another (for example, due to an application error).

To perform the HARG switchover, run the following command:

`/opt/OV/bin/OpC/opchamgr -online <HARG_name> <second_node>`

After you run this command, the HARG is switched between the nodes (that is, the HARG becomes `ONLINE` on the selected node).

Failover Scenario

This example shows the failover scenario in which the failover occurs because the management server on the node on which the HARG is active stops running. After a few moments, the HA Manager detects that the HP Operations management server does not run and switches the HARG to a second node. The HARG status on the node on which the HP Operations management server is down is marked as `FAULTED`. This means that there is a problem with the HARG on this node and it is not possible to switch the HARG to this node. Make sure that you fix the problem with the HARG. For example, you can start the HP Operations management server and clear the `FAULTED` flag for the HARG by running the following command:

`/opt/OV/bin/OpC/opchamgr -clear <HARG_name> [<node_where_HARG_is_FAULTED>]`

If the HARG is `FAULTED` on both nodes and the `FAULTED` flag is cleared, the HARG is automatically started on the primary node within one minute.

To monitor the HA Manager status, you can use the following while loop:

```
# while true
# do
#   /opt/OV/bin/OpC/opchamgr -status > /HAMGR_status.txt
#   clear
#   cat /HAMGR_status.txt
#   sleep 2
# done
```

HA Manager Status

To check the HA Manager status, run the following command:

`/opt/OV/bin/OpC/opchamgr -status`

The following is an example output of the HA Manager status:

```
Nodes :
=======
nodeA : ONLINE
nodeB : ONLINE
HARGS :
=======
hpom-server :
 General status : ONLINE
 Active node    : nodeA
 Primary node   : nodeA
 Failover node  : nodeB
 ---------------
 nodeA : ONLINE
 nodeB : OFFLINE
```

The status values used for the node and HARG statuses are `ONLINE`, `OFFLINE`, `FAULTED`, and `DISABLED`.

Log Files and Trace Files

When troubleshooting, you can use a log file analysis that represents a useful methodology for understanding all the aspects of the HA Manager feature. To help you investigate the cause of problems, you can also use problem tracing. Trace and log files can help you pinpoint when and where problems occurred.

HARG Log and Trace Files

All errors performed during the HARG online, offline, or monitor procedure are written into the `error.log` file that you can find at the following location:

`/var/opt/OV/hacluster/<HARG_name>`

To enable HARG tracing on a local node, run the following command:

`/opt/OV/lbin/ovharg -tracing <HARG_name> enable`

Trace information is stored in the `trace.log` file that you can find at the following location:

`/var/opt/OV/hacluster/<HARG_name>`

If you want to disable HARG tracing on a local node, run the following command:

`/opt/OV/lbin/ovharg -tracing <HARG_name> disable`

HA Manager Trace File

To enable HA Manager tracing on a local node, run the following command:

```
/opt/OV/bin/OpC/opchamgr -trace enable
```

Trace information is stored in the `hamgr-trace.log` file that you can find at the following location:

```
/var/opt/OV/hamanager
```

To disable HA Manager tracing on a local node, run the following command:

```
/opt/OV/bin/OpC/opchamgr -trace disable
```

Data Flow

This section contains several examples of the data flow related to the HA Manager and the HARG, so that you can see how to address the issues shown by these examples.

Starting the HA Manager

When the HA Manager is started, it forwards the local status to all nodes. After some time, the HA Manager starts the HARG on a local node if the local node is set as a primary node.

Performing a HARG Switchover

This use case shows which steps are performed during a HARG switchover.

The following is the status before the HARG switchover:

```
Nodes :
=======
nodeA : ONLINE
nodeB : ONLINE

HARGS :
=======
hpom-server :
 General status : ONLINE
 Active node : nodeA
 Primary node : nodeA
 Failover node : nodeB
 ---------------
 nodeA : ONLINE
 nodeB : OFFLINE
```

The HARG switchover procedure consists of the following steps:

1. To make the HARG online on a selected node, run the following command:

   ```
   /opt/OV/bin/Opc/opchamgr -online hpom-server nodeB
   ```

2. On `nodeA`, the HA Manager runs the following command:

   ```
   /opt/OV/lbin/ovharg -stop hpom-server
   ```

3. The ovharg command executes all K* tools in the /var/opt/OV/hacluster/hpom-server directory.
4. The HA Manager on nodeA informs the HA Manager on nodeB to start the HARG.
5. On nodeB, the HA Manager runs the following command:

   /opt/OV/lbin/ovharg -start hpom-server
6. The ovharg command executes all S* tools in the /var/opt/OV/hacluster/hpom-server directory.
7. When the ovharg command successfully executes all S* tools, the HARG becomes ONLINE.

The following is the status after the HARG switchover:

```
Nodes :
=======
nodeA : ONLINE
nodeB : ONLINE

HARGS :
=======
hpom-server :
 General status : ONLINE
 Active node : nodeB
 Primary node : nodeA
 Failover node : nodeA
 ---------------
 nodeA : OFFLINE
 nodeB : ONLINE
```

Stopping a HARG Manually

This use case shows which steps are performed to stop a HARG manually.

The following is the status before the HARG is stopped:

```
Nodes :
=======
nodeA : ONLINE
nodeB : ONLINE

HARGS :
=======
hpom-server :
 General status : ONLINE
 Active node : nodeA
 Primary node : nodeA
 Failover node : nodeB
 ---------------
 nodeA : ONLINE
 nodeB : OFFLINE
```

When stopping the HARG manually, the following steps are performed:

1. To make the HARG offline on a selected node, run the following command:

   `/opt/OV/bin/OpC/opchamgr -offline hpom-server`
2. On `nodeA`, the HA Manager runs the following command:

   `/opt/OV/lbin/ovharg -stop hpom-server`
3. The `ovharg` command executes all `K*` tools in the `/var/opt/OV/hacluster/hpom-server` directory.

The following is the status after the HARG is stopped:

```
Nodes :
=======
nodeA : ONLINE
nodeB : ONLINE

HARGS :
=======
hpom-server :
 General status : OFFLINE
 Active node :
 Primary node : nodeA
 ----------------
 nodeA : OFFLINE
 nodeB : OFFLINE
```

Automatic HARG Failover

This use case shows what happens during an automatic HARG failover.

The following is the status before the HARG failover:

```
Nodes :
=======
nodeA : ONLINE
nodeB : ONLINE

HARGS :
=======
hpom-server :
 General status : ONLINE
 Active node : nodeA
 Primary node : nodeA
 Failover node : nodeB
 ---------------
 nodeA : ONLINE
 nodeB : OFFLINE
```

The automatic HARG failover procedure consists of the following steps:

1. The HA Manager checks the HARG status by running the following command:

   `/opt/OV/lbin/ovharg -monitor hpom-server`
2. The `ovharg` command executes all `M*` tools in the `/var/opt/OV/hacluster/hpom-server` directory.
3. One of `M*` tools returns an error code 1.
4. The `ovharg` command returns an error code 1. This is the information for HA Manager that there is something wrong with the HARG.
5. The HA Manager switches the HARG to the failover node (that is, `nodeB`).
6. On `nodeA`, the HA Manager runs the following command:

   `/opt/OV/lbin/ovharg -stop hpom-server`
7. The `ovharg` command executes all `K*` tools in the `/var/opt/OV/hacluster/hpom-server` directory.
8. The HA Manager on `nodeA` informs the HA Manager on `nodeB` to start the HARG.
9. On `nodeB`, the HA Manager runs the following command:

   `/opt/OV/lbin/ovharg -start hpom-server`
10. The `ovharg` command executes all `S*` tools in the `/var/opt/OV/hacluster/hpom-server` directory.
11. When the `ovharg` command successfully executes all `S*` tools, the HARG becomes `ONLINE`.

The following is the status after the HARG failover:

```
Nodes :
=======
nodeA : ONLINE
nodeB : ONLINE

HARGS :
=======
hpom-server :
 General status : FAULTED
 Active node : nodeB
 Primary node : nodeA
 ---------------
 nodeA : FAULTED
 nodeB : ONLINE
```

FAULTED nodeB

This use case shows what happens when `nodeB` becomes `FAULTED`.

The following is the status before `nodeB` becomes `FAULTED`:

```
Nodes :
=======
```

```
nodeA : ONLINE
nodeB : ONLINE
HARGS :
=======
hpom-server :
 General status : ONLINE
 Active node : nodeB
 Primary node : nodeA
 Failover node : nodeA
 ---------------
 nodeA : OFFLINE
 nodeB : ONLINE
```

When nodeb becomes FAULTED, the following steps occur:

1. The HA Manager on nodeB detects that the reference node is down and there is no status update from nodeA for some time. Therefore, the HA Manager assumes that there is something wrong with nodeB and marks nodeB as FAULTED.
2. Because nodeB is FAULTED, the HA Manager stops all HARGs that are ONLINE on this node.
3. On nodeB, the HA Manager runs the following command:

   ```
   /opt/OV/lbin/ovharg -stop hpom-server
   ```
4. The ovharg command executes all K* tools in the /var/opt/OV/hacluster/hpom-server directory.
5. At this point, the status reported on nodeB is the following:

   ```
   Nodes :
   =======
   nodeA : OFFLINE
   nodeB : FAULTED

   HARGS :
   =======
   hpom-server :
    General status : OFFLINE
    Active node :
    Primary node : nodeA
    ---------------
    nodeA : OFFLINE
    nodeB : OFFLINE
   ```
6. The HA Manager on nodeA detects that there is no status update from nodeB for some time. Therefore, the HA Manager assumes that there is something wrong with nodeB and marks nodeB as FAULTED.
7. Because nodeB is FAULTED and the HARG was ONLINE on nodeB, the HA Manager starts the HARG on nodeA, which is the primary node.
8. On nodeA, the HA Manager runs the following command:

```
/opt/OV/lbin/ovharg -start hpom-server
```

9. The `ovharg` command executes all `S*` tools in the `/var/opt/OV/hacluster/hpom-server` directory.
10. When the `ovharg` command successfully executes all `S*` tools, the HARG becomes `ONLINE`.

The status reported on `nodeA` is the following:

```
Nodes :
=======
nodeA : ONLINE
nodeB : FAULTED

HARGS :
=======
hpom-server :
 General status : ONLINE
 Active node : nodeA
 Primary node : nodeA
 ---------------
 nodeA : ONLINE
 nodeB : OFFLINE
```

Troubleshooting

This section describes solutions to the specific problems you may encounter:

- "HARG Status Is FAULTED" on page 753

  What to do if the status report shows that the HARG status is `FAULTED`.
- "Node Status Is FAULTED" on page 755

  What to do if the status report shows that the node status is `FAULTED`.

HARG Status Is FAULTED

Problem

The status report shows the following:

```
Nodes :
=======
nodeA : ONLINE
nodeB : ONLINE

HARGS :
=======
hpom-server :
 General status : ONLINE
 Active node : nodeA
 Primary node : nodeA
```

```
--------------
nodeA : ONLINE
nodeB : FAULTED
```

Solution

To solve the problem, follow these steps:

1. On `nodeB`, check the HARG error file:

   `/var/opt/OV/hacluster/<HARG_name>/error.log`

   The log file shows which monitor tool reported an error and it may also contain detailed information about the problem.
2. Depending on the monitor tool reporting a problem, a further investigation involving system administration must be performed.
3. When you find the cause of the problem and fix it, you can clear the `FAULTED` flag by using the `opchamgr -clear <HARG_name>` command.

If this is a repeated problem, follow these steps:

1. Make the HARG offline by running the following command:

   `/opt/OV/bin/OpC/opchamgr -offline <HARG_name>`
2. Disable the HARG by running the following command:

   `/opt/OV/bin/OpC/opchamgr -harg -disable <HARG_name>`
3. On the problematic node, manually start the HARG in the same way as the HA Manager starts it, that is, by running the following command:

   `/opt/OV/lbin/ovharg -start <HARG_name>`

   The return code should be 0.
4. Check the HARG status regularly by running the following command:

   `/opt/OV/lbin/ovharg -monitor <HARG_name>`

   Besides the status check, check also other system parameters that are useful for problem detection.

   When the HARG runs properly, the return code is 0. Otherwise, the return code is 1.
5. When the HARG status reports a problem (that is, when the return code is 1), analyze all provided data, find the cause of the problem, and then solve it.
6. Stop the HARG by running the following command:

   `/opt/OV/lbin/ovharg -stop <HARG_name>`
7. Enable the HARG by running the following command:

   `/opt/OV/bin/OpC/opchamgr -harg -enable <HARG_name>`

   After a few moments, the HARG is automatically started on the primary node.

Node Status Is FAULTED

Problem

The status report shows the following:

```
Nodes :
=======
nodeA : ONLINE
nodeB : FAULTED

HARGS :
=======
hpom-server :
 General status : ONLINE
 Active node    : nodeA
 Primary node   : nodeA
 ---------------
 nodeA : ONLINE
 nodeB : OFFLINE
```

Solution

To solve the problem, follow these steps:

1. Check if all nodes are reachable by using the `ping` command, and then try to log on to problematic nodes.
2. Check if the HA Manager runs on all nodes by using the `opchamgr -daemon` command.
3. On each node, check if the remote shell or secure remote shell communication with the other nodes works (for example, try to copy several files between the nodes by using the remote shell copy).

Appendices

The appendices in this document provide additional information that you may require for administration and maintenance of your HPOM managed environment.

For details, see:

Appendix A: HPOM Managed Node APIs and Libraries

In this Appendix

This chapter provides information about the application-programming interfaces (APIs) that HPOM provides. For example, HPOM allows applications to use the application-programming interface to automatically provide monitor values to HPOM or submit a message.

The information in this section covers the following topics:

HPOM APIs on Managed Nodes

Table 66 describes commands associated with application program interfaces (APIs) on HPOM managed nodes.

Table 66: HPOM APIs on Managed Nodes

API	Command	Description
N/A	`opcmack(1)`	Acknowledges an HPOM message received from the message agent on the managed node and sent to the management server.
`opcmon(3)`	`opcmon(1)`	Sends the current value of a monitored object to the HPOM monitoring agent on the local managed node.
`opcmsg(3)`	`opcmsg(1)`	Submits a message to the HPOM message interceptor on the local managed node.

For more detailed information about the commands listed in Table 66, including the parameters and options that are available, see the respective manual pages for the commands as described in "Manual Pages" on page 782.

An example of how the API functions are used is available in the following file on the management server:

`/opt/OV/OpC/examples/progs/opcapitest.c`

For the corresponding makefiles, see the HP Operations agent documentation.

HPOM Managed Node Libraries

HPOM C functions are available in a shared library. The definitions and return values for the functions are defined in the HPOM include file, `opcapi.h`. For more information about the location of the include file, the required libraries and the makefile for a specific managed node platform, see the HP Operations agent documentation.

Note: Customer applications must be linked to HPOM using the libraries provided, as well as the link and compile options, in the HP Operations agent documentation. Integration is only supported if applications are linked.

An example of how the API functions are used is available in the following file on the management server:

`/opt/OV/OpC/examples/progs/opcapitest.c`

This directory also contains the makefiles for building the examples. These makefiles use the compile and link options needed to correctly build an executable.

Appendix B: HPOM Database Tables and Tablespaces

In this Appendix

This appendix describes the tables and tablespaces that HPOM uses in the in databases for example, to store messages, message annotations, managed node names, and so on. For detailed information about the function of the HPOM tables in the Relational Database Management System (RDBMS), see the *HPOM Reporting and Database Schema*.

The information in this section covers the following topics:

HPOM Tables and Tablespaces in a Database

A database uses tablespaces to manage available disk space. You can assign datafiles of a fixed size to tablespaces. In Oracle, the size of the various datafiles assigned to a tablespace determines the size of the tablespace. Table 67 shows the default tablespace design and the assigned database tables for HPOM-related data. To increase the size of a tablespace, you must add a datafile of a particular size to the tablespace. You can add datafiles interactively using the Oracle tool, Server Manager, or using the following `sql` command: `alter tablespace add datafile`.

Note: In PostgreSQL, tablespaces do not have a predetermined size. The size of the tablespaces depends on available space of the media where the tablespaces are located.

The PostgreSQL tablespaces use less data than the Oracle ones. However, make sure that enough disk space is available.

Table 67: HPOM Tables and Tablespaces in a Database

Tables	Tablespace	Oracle Size	Comments
`opc_act_ messages`	`OPC_1`	`SIZE 4M AUTOEXTEND ON NEXT 6M MAXSIZE 500M` `DEFAULT STORAGE (` `INITIAL 2M`	Tables with a heavy load. Indexes are not on the same disk as the table, thus providing extra tablespace.

HPOM Tables and Tablespaces in a Database , continued

Tables	Tablespace	Oracle Size	Comments
		NEXT 2M PCTINCREASE 0)	
opc_anno_text opc_annotation opc_msg_text opc_orig_msg_text	OPC_2	SIZE 5M AUTOEXTEND ON NEXT 6M MAXSIZE 500M DEFAULT STORAGE (INITIAL 1M NEXT 1M PCTINCREASE 0)	Tables with a heavy load. Indexes are not on the same disk as the table, thus providing extra tablespace.
opc_node_names	OPC_3	SIZE 1M AUTOEXTEND ON NEXT 1M MAXSIZE 500M DEFAULT STORAGE (INITIAL 256K NEXT 256K PCTINCREASE 0)	Table with very frequent access.
All other tables	OPC_4	SIZE 26M AUTOEXTEND ON NEXT 2M MAXSIZE 340M DEFAULT STORAGE (INITIAL 64K NEXT 1M PCTINCREASE 0)	None.
Default tablespace of user opc_op	OPC_5	SIZE 1M AUTOEXTEND ON NEXT 1M MAXSIZE 500M DEFAULT STORAGE (INITIAL 32K NEXT 1M PCTINCREASE 0)	None.

HPOM Tables and Tablespaces in a Database , continued

Tables	Tablespace	Oracle Size	Comments
opc_hist_ messages	OPC_6	SIZE 4M AUTOEXTEND ON NEXT 2M MAXSIZE 500M DEFAULT STORAGE (INITIAL 2M NEXT 2M PCTINCREASE 0)	Tables with a heavy load. Indexes are not on the same disk as the table, thus providing extra tablespace.
opc_hist_msg_ text	OPC_7	SIZE 4M AUTOEXTEND ON NEXT 2M MAXSIZE 500M DEFAULT STORAGE (INITIAL 2M NEXT 2M PCTINCREASE 0)	Tables with a heavy load. Indexes are not on the same disk as the table, thus providing extra tablespace.
opc_hist_orig_ text	OPC_8	SIZE 4M AUTOEXTEND ON NEXT 2M MAXSIZE 500M DEFAULT STORAGE (INITIAL 2M NEXT 2M PCTINCREASE 0)	Tables with a heavy load. Indexes are not on the same disk as the table, thus providing extra tablespace.
opc_hist_ annotation opc_hist_anno_ text	OPC_9	SIZE 6M AUTOEXTEND ON NEXT 2M MAXSIZE 500M DEFAULT STORAGE (INITIAL 2M NEXT 2M PCTINCREASE 0)	Tables with a heavy load. Indexes are not on the same disk as the table, thus providing extra tablespace.

HPOM Tables and Tablespaces in a Database , continued

Tables	Tablespace	Oracle Size	Comments
opc_service_log opc_service	OPC_10	SIZE 6M AUTOEXTEND ON NEXT 6M MAXSIZE 500M DEFAULT STORAGE (INITIAL 2M NEXT 2M PCTINCREASE 0)	Tables with a heavy load. Indexes are not on the same disk as the table, thus providing extra tablespace.
Temporary data (used for sorting)	OPC_TEMP	SIZE 1M AUTOEXTEND ON NEXT 1M MAXSIZE 500M DEFAULT STORAGE (INITIAL 512K NEXT 512K PCTINCREASE 0)	None.
Index tablespace for active messages	OPC_INDEX1	SIZE 13M AUTOEXTEND ON NEXT 1M MAXSIZE 500M DEFAULT STORAGE (INITIAL 1M NEXT 1M PCTINCREASE 0)	Disk other than than for the following tablespaces: opc_act_messages
Index tablespace for history messages	OPC_INDEX2	SIZE 10M AUTOEXTEND ON NEXT 1M MAXSIZE 500M DEFAULT STORAGE (INITIAL 1M NEXT 1M PCTINCREASE 0)	Disk other than that for the following tablespaces: opc_hist_messages
Index tablespace for service logging	OPC_INDEX3	SIZE 10M AUTOEXTEND ON NEXT 1M MAXSIZE 500M	Disk other than for the following tablespaces:

HPOM Tables and Tablespaces in a Database , continued

Tables	Tablespace	Oracle Size	Comments
		DEFAULT STORAGE (INITIAL 1M NEXT 1M PCTINCREASE 0)	opc_service_log

Non-HPOM Tables and Tablespaces

Note: This section only applies if you use an Oracle database.

Table 68 lists the non-HPOM tablespaces in an Oracle database.

Table 68: Non-HPOM Tablespaces

Tables	Tablespace	Size	Comments
System tables	SYSTEM	SIZE 50M DEFAULT STORAGE (INITIAL 16K NEXT 16K PCTINCREASE 50)	None
Temporary data	TEMP	SIZE 2M AUTOEXTEND ON NEXT 1M MAXSIZE 500M DEFAULT STORAGE (INITIAL 100K NEXT 100K PCTINCREASE 0)	None
Rollback segments	RBS1	SIZE 1M AUTOEXTEND ON NEXT 1M MAXSIZE 500M DEFAULT STORAGE (INITIAL 500K	Tablespace with a heavy load

Non-HPOM Tablespaces , continued

Tables	Tablespace	Size	Comments
		NEXT 500K MINEXTENTS 10 PCTINCREASE 0)	
Tablespace for Oracle Tool Tables (for example, Report Writer)	TOOLS	SIZE 1M AUTOEXTEND ON NEXT 1M MAXSIZE 100M DEFAULT STORAGE (INITIAL 100K NEXT 100K PCTINCREASE 0)	None

Appendix C: HPOM Audits

In this Appendix

The information in this appendix lists the areas of HP Operations Manager that you can target for audit, describes the actions and operations that produce an entry in the audit log files, and indicates the default level of information logged when the action occurs. For example, you can find out how to monitor changes made to user configurations or any HPOM objects. You can also learn how to use the audit facility to monitor when the scripts and binaries that HPOM uses are started and stopped and if any changes occur to the HPOM processes used by the management server and managed nodes.

In the area of user configuration, you can monitor the names of users who log on and out, when the logons occur, what profiles are assigned to user, and what, if any, changes occur to the user or user profile.

You can also enable auditing for individual HPOM objects such as: managed nodes, node groups, policies, or messages. Any attempts to upload or download configuration data can be logged, too.

You can monitor the use of HPOM scripts, and binaries, for example, when the command-line interface to HPOM is used, or a license check fails.

Finally, you can audit the HPOM processes that control the management server and managed nodes, for example, when the processes are started or stopped.

HPOM Audit Areas

The information in this section explains how to set up an HPOM audit and how to use it to monitor the HPOM environment. The information covers the following areas:

- HPOM user security:

  For more information about the various aspects of user security and authorization that you can target for auditing in HPOM, see "HPOM User Audits" on page 766.

- HPOM objects:

  For more information about the HPOM objects that you can target for auditing, see "HPOM Object Audit Areas" on page 770.

- HPOM scripts and binaries:

  For more information about the HPOM scripts and binaries that you can target for auditing, see "HPOM Scripts and Binaries" on page 780.

- HPOM processes:

For more information about HPOM processes that you can target for auditing, see "HPOM Processes" on page 780.

Note that the default audit level signifies the importance that HPOM attaches by default to a particular audit area. You can use this level to control the amount of information that HPOM writes in audit log files. For example, if you set the audit level to “MAJOR”, all actions tagged with the audit level “MAJOR” and anything that is less important (MINOR) are logged in the audit trace files.

HPOM User Audits

This section describes the aspects of HPOM users that you can target for auditing and indicates the audit level that is set by default. The information included in this section covers the following areas:

- "HPOM User-Logons Audit" on page 766
- "HPOM User-Configuration Audit " on page 767
- "HPOM User-Profile Audit" on page 768
- "HPOM Security-Certificate Audit " on page 769

Table 69 lists the HPOM objects that you can target for auditing in the area of user logons and logouts. Note that, for auditing purposes, the tracing of successful user logons and logouts is disabled by default. If you want HPOM to write entries in the audit log files for successful user logons and logouts, change the audit level, for example, to “MINOR” or “MAJOR”.

Table 69: HPOM User-Logons Audit

User Audit Area	Use Case	Default Audit Level [a]	ovoconf Variable in Audit Name Space
Login	User logon succeeds	Disabled	LOGIN_SUCCESS
Login	User logon fails	MAJOR	LOGIN_FAILURE
Login	User logon succeeds from the Java GUI - Client process	SERIOUS	LOGIN_SUCCESS
Login	User logon succeeds from the Java GUI	SERIOUS	LOGIN_SUCCESS
Login	User logon fails from the Java GUI	MAJOR	LOGIN_FAILURE
Logout	User connection	Disabled	LOGOUT

[a]Minor, Major, Serious, or Internal

HPOM User-Logons Audit, continued

User Audit Area	Use Case	Default Audit Level [a]	ovoconf Variable in Audit Name Space
	closes		
Logout	User logs out from the Java GUI - Client process	MAJOR	LOGOUT
Logout	User logs out from the Java GUI	MAJOR	LOGOUT

Table 70 lists the actions and operations that you can target for auditing in the area of user configuration.

Table 70: HPOM User-Configuration Audit

User Audit Area	Use Case	Default Audit Level [b]	ovoconf Variable in Audit Name Space
User	Administrator modifies user	SERIOUS	OM_CFG_CHG_USER
User	Administrator deletes user	MAJOR	OM_CFG_DEL_USER
User	Administrator assigns user responsibility	SERIOUS	OM_CFG_USER_RESP
User	Administrator deassigns user responsibility	SERIOUS	OM_CFG_USER_RESP
User	Administrator changes a user password	SERIOUS	OM_CFG_USER_PWD_CHANGE
User	Administrator creates a user with administrator privileges	SERIOUS	OM_CFG_ADD_USER
User	Administrator creates a new user	MAJOR	OM_CFG_ADD_USER

[a] Minor, Major, Serious, or Internal

[b] Minor, Major, Serious, or Internal

HPOM User-Configuration Audit , continued

User Audit Area	Use Case	Default Audit Level [a]	ovoconf Variable in Audit Name Space
User	Administrator assigns a user profile to a user or a user profile	SERIOUS	OM_CFG_CHG_USER
User	Administrator deassigns a user profile from a user or user profile	SERIOUS	OM_CFG_CHG_USER
User	Administrator assigns an application to a user	MINOR	OM_CFG_CHG_USER
User	Administrator deassigns an application from a user	MINOR	OM_CFG_CHG_USER
User	Administrator assigns an application group to a user	MINOR	OM_CFG_CHG_USER
User	Administrator deassigns an application group from a user	MINOR	OM_CFG_CHG_USER

Table 71 lists the HPOM actions that you can target for auditing in the area of user profiles.

Table 71: HPOM User-Profile Audit

User Audit Area	Use Case	Default Audit Level [b]	ovoconf Variable in Audit Name Space
User Profile	Administrator creates a user profile	MINOR	OM_CFG_ADD_USER_PROFILE
User Profile	Administrator copies a user profile	MINOR	none
User Profile	Administrator modifies	MINOR	OM_CFG_CHG_USER_

[a]Minor, Major, Serious, or Internal

[b]Minor, Major, Serious, or Internal

HPOM User-Profile Audit, continued

User Audit Area	Use Case	Default Audit Level [a]	ovoconf Variable in Audit Name Space
	a user profile		PROFILE
User Profile	Administrator deletes a user profile	MAJOR	OM_CFG_DEL_USER_PROFILE
User Profile	Administrator assigns an application to a user profile	MINOR	OM_CFG_CHG_PROFILE
User Profile	Administrator deassigns an application from a user profile	MINOR	OM_CFG_CHG_PROFILE
User Profile	Administrator assigns an application group to a user profile	MINOR	OM_CFG_CHG_PROFILE
User Profile	Administrator deassigns an application group from a user profile	MINOR	OM_CFG_CHG_PROFILE

Table 72 lists the HPOM objects that you can target for auditing in the area of security certificates.

Table 72: HPOM Security-Certificate Audit

User Audit Area	Use Case	Default Audit Level [b]	ovoconf Variable in Audit Name Space
Certificate	New certificate request created	MAJOR	OM_SV_REQUEST_CERTIFICATE
Certificate	Certificate request granted	SERIOUS	OM_SV_GRANT_CERT_REQUEST
Certificate	Certificate request denied	MAJOR	OM_SV_DENY_CERT_REQUEST

[a] Minor, Major, Serious, or Internal

[b] Minor, Major, Serious, or Internal

HPOM Security-Certificate Audit , continued

User Audit Area	Use Case	Default Audit Level [a]	ovoconf Variable in Audit Name Space
Certificate	Certificate request deleted	MAJOR	OM_SV_DEL_CERT_REQUEST
Certificate	Generic certificate event occurs	MINOR	none

HPOM Object Audit Areas

Table 73 lists the actions and operations that you can target for auditing in the area of HPOM objects.

Table 73: HPOM-Object Audit Areas

Object Audit Area	Use Case	Default Audit Level [b]	ovoconf Variable in Audit Name Space
HPOM Message	User owns a message	MINOR	OM_MESSAGE_OWN
HPOM Message	User disowns a message	MINOR	OM_MESSAGE_OWN
HPOM Message	User forwards a message to the Trouble Ticket interface	MINOR	OM_MSG_FWD_NS_IF
HPOM Message	User forwards a message to the Notification Service interface	MINOR	OM_MSG_FWD_TT_IF
HPOM Message	User deletes one or more HPOM messages	MINOR	OM_MSG_DEL
HPOM Message	User acknowledges one or more HPOM messages	MINOR	OM_MSG_MULTI_ACK

[a]Minor, Major, Serious, or Internal

[b]Minor, Major, Serious, or Internal

HPOM-Object Audit Areas , continued

Object Audit Area	Use Case	Default Audit Level [a]	ovoconf Variable in Audit Name Space
HPOM Message	User adds an annotation to a message	MINOR	OM_CFG_ADD_ ANNOTATION
HPOM Message	User modifies an annotation	MINOR	OM_CFG_CHG_ ANNOTATION
HPOM Message	User removes an annotation from a message	MINOR	OM_CFG_DEL_ ANNOTATION
HPOM Node	User creates a node	MINOR	OM_CFG_ADD_NODE
HPOM Node	User modifies a node	MINOR	OM_CFG_CHG_NODE
HPOM Node	User deletes a node	MAJOR	OM_CFG_DEL_NODE
HPOM Node	User assigns a policy to a node	MINOR	OM_CFG_CHG_NODE
HPOM Node	User deassigns a policy from a node	MINOR	OM_CFG_CHG_NODE
HPOM Node	User assigns a policy to a node group	MINOR	OM_CFG_CHG_NODE_GRP
HPOM Node	User deassigns a policy from a node group	MINOR	OM_CFG_CHG_NODE_GRP
HPOM Node	User assigns a policy group to a node	MINOR	OM_CFG_CHG_NODE
HPOM Node	User deassigns a policy group from a node	MINOR	OM_CFG_CHG_NODE
HPOM Node	User assigns a policy group to a node group	MINOR	OM_CFG_CHG_NODE_GRP

[a]Minor, Major, Serious, or Internal

HPOM-Object Audit Areas , continued

Object Audit Area	Use Case	Default Audit Level [a]	ovoconf Variable in Audit Name Space
HPOM Node	User deassigns a policy group from a node group	MINOR	OM_CFG_CHG_NODE_GRP
HPOM Node	User assigns a category to a node	MINOR	OM_CFG_CHG_NODE
HPOM Node	User deassigns a category a the node	MINOR	OM_CFG_CHG_NODE
HPOM Node	User installs a subagent	SERIOUS	OM_SUBAGT_INSTALL
HPOM Node	User removes a subagent	SERIOUS	OM_SUBAGT_DEINSTALL
HPOM Node	User reinstalls a subagent	SERIOUS	OM_SUBAGT_INSTALL
HPOM Node	User activates a subagent	SERIOUS	OM_SUBAGT_INSTALL
HPOM Node	User deploys agent software to a node	SERIOUS	OM_AGT_SW_INSTALL
HPOM Node	User removes agent software from a node	SERIOUS	OM_AGT_SW_DEINSTALL
HPOM Node	User updates the flexible-management policy deployment	SERIOUS	OM_AGT_MGRCONF_ DEPLOY
HPOM Node	User deploys HPOM instrumentation to a managed node	SERIOUS	OM_AGT_INSTR_DEPLOY
HPOM Application	User starts a terminal application	MAJOR	OM_TERMINAL_APP_ LAUNCH

[a]Minor, Major, Serious, or Internal

HPOM-Object Audit Areas , continued

Object Audit Area	Use Case	Default Audit Level [a]	ovoconf Variable in Audit Name Space
HPOM Application	User starts an application	MAJOR	OM_APP_START
HPOM Application	User creates an HPOM application	MINOR	OM_CFG_ADD_APPL
HPOM Application	User modifies an HPOM application	MINOR	OM_CFG_CHG_APPL
HPOM Application	User deletes an HPOM application	MAJOR	OM_CFG_DEL_APPL
HPOM External Application	User registers an external application with the message-stream interface (MSI)	MINOR	OM_SV_REGISTER_MSI
HPOM External Application	User unregisters an external application from the MSI	MINOR	OM_SV_REGISTER_MSI
HPOM Config	User downloads messages from the history browser	MINOR	OM_SV_HIST_MSG_ DOWNLOAD
HPOM Config	User uploads messages to the history browser	SERIOUS	OM_SV_HIST_MSG_ UPLOAD
HPOM Config	User performs a configuration upload	SERIOUS	OM_CFG_UPLOAD
HPOM Config	User modifies the database-maintenance configuration	SERIOUS	OM_CFG_DB_ MAINTENANCE
HPOM Config	User modifies the management-server	SERIOUS	OM_CFG_DB

[a]Minor, Major, Serious, or Internal

HPOM-Object Audit Areas , continued

Object Audit Area	Use Case	Default Audit Level [a]	ovoconf Variable in Audit Name Space
	configuration		
HPOM Config	User performs a configuration download	MINOR	OM_CFG_DOWNLOAD
HPOM Config	Administrator activates heartbeat monitoring for a managed node	SERIOUS	OM_CFG_CHG_NODE_HEARTBEAT
HPOM Config	Administrator deactivates heartbeat monitoring for a node	SERIOUS	OM_CFG_CHG_NODE_HEARTBEAT
HPOM Config	Administrator changes the heartbeat monitoring interval for a node	SERIOUS	OM_CFG_CHG_NODE_HEARTBEAT
HPOM Config	User uploads generic policies from a directory	MAJOR	OM_CFG_UPLOAD_POLICY_TYPE
HPOM Config	User uploads generic policies from a file	MINOR	OM_CFG_UPLOAD_POLICY_TYPE
HPOM Config	User enables duplicate message suppression	SERIOUS	OM_CFG_DUP_MSG_SUPPRESS
HPOM Config	User disables duplicate message suppression	SERIOUS	OM_CFG_DUP_MSG_SUPPRESS
HPOM Config	User changes global options for the HPOM management server: parallel distribution	SERIOUS	OM_CFG_MISC
HPOM Config	HPOM reads Service	MAJOR	OM_CFG_READ_SERVNAV

[a]Minor, Major, Serious, or Internal

HPOM-Object Audit Areas , continued

Object Audit Area	Use Case	Default Audit Level [a]	ovoconf Variable in Audit Name Space
	Navigator configuration		
HPOM Config	HPOM notices a change to the Service Navigator configuration	SERIOUS	OM_CFG_WRITE_SERVNAV
HPOM Config	User performs a backup	MINOR	OM_SV_BACKUP
HPOM Config	User enables customized startup message	MINOR	OM_STARTUPMSG
HPOM Config	User disables customized startup message	MINOR	OM_STARTUPMSG
HPOM Config	User modifies customized startup message	MINOR	OM_STARTUPMSG
HPOM Config	User deletes customized startup message	MINOR	OM_STARTUPMSG
HPOM Other	User creates a new category	MINOR	OM_CFG_ADD_CATEGORY
HPOM Other	User modifies a category	MINOR	OM_CFG_CHG_CATEGORY
HPOM Other	User deletes a category	MAJOR	OM_CFG_DEL_CATEGORY
HPOM Other	User registers a generic policy type	MINOR	OM_CFG_ADD_POLICY_TYPE
HPOM Other	User modifies a generic policy type	MINOR	OM_CFG_CHG_POLICY_TYPE

[a]Minor, Major, Serious, or Internal

HPOM-Object Audit Areas , continued

Object Audit Area	Use Case	Default Audit Level [a]	ovoconf Variable in Audit Name Space
HPOM Other	User deletes a generic policy type	MAJOR	OM_CFG_DEL_POLICY_TYPE
HPOM Other	User creates an application group	MINOR	OM_CFG_ADD_APPL_GRP
HPOM Other	User modifies an application group	MINOR	OM_CFG_CHG_APPL_GRP
HPOM Other	User deletes an application group	MAJOR	OM_CFG_DEL_APPL_GRP
HPOM Other	User assigns an application to an application group	MINOR	OM_CFG_CHG_APPL_GRP
HPOM Other	User deassigns an application from an application group	MINOR	OM_CFG_CHG_APPL_GRP
HPOM Other	User assigns an application group to another application group	MINOR	OM_CFG_CHG_APPL_GRP
HPOM Other	User deassigns an application group from another application group	MINOR	OM_CFG_CHG_APPL_GRP
HPOM Other	User creates a condition	MINOR	OM_CFG_CHG_POLICY
HPOM Other	User deletes a condition	MINOR	OM_CFG_CHG_POLICY
HPOM Other	User adds/sets an instruction interface	MINOR	OM_CFG_DEL_INSTR_IF

[a]Minor, Major, Serious, or Internal

HPOM-Object Audit Areas , continued

Object Audit Area	Use Case	Default Audit Level [a]	ovoconf Variable in Audit Name Space
HPOM Other	User copies an instruction interface	MINOR	OM_CFG_CPY_INSTR_IF
HPOM Other	User modifies an instruction interface	MINOR	OM_CFG_CHG_INSTR_IF
HPOM Other	User deletes an instruction interface	MAJOR	OM_CFG_DEL_INSTR_IF
HPOM Other	User creates a new message group	MINOR	OM_CFG_ADD_MSG_GRP
HPOM Other	User modifies a message group	MINOR	OM_CFG_CHG_MSG_GRP
HPOM Other	User modifies a message-group name	SERIOUS	OM_CFG_CHG_MSG_GRP
HPOM Other	User deletes a message group	MAJOR	OM_CFG_DEL_MSG_GRP
HPOM Other	User creates a node-layout hierarchy	MINOR	OM_CFG_NODE_LAYOUT
HPOM Other	User modifies a node-layout hierarchy	MINOR	OM_CFG_NODE_LAYOUT
HPOM Other	User deletes a node-layout hierarchy	MAJOR	OM_CFG_NODE_LAYOUT
HPOM Other	User creates a layout group	MINOR	OM_CFG_LAYOUT_GRP
HPOM Other	User modifies a layout group	MINOR	OM_CFG_LAYOUT_GRP
HPOM Other	User deletes a layout group	MAJOR	OM_CFG_LAYOUT_GRP
HPOM Other	User creates a	MINOR	OM_CFG_CHG_NOTIFY_

[a]Minor, Major, Serious, or Internal

HPOM-Object Audit Areas , continued

Object Audit Area	Use Case	Default Audit Level [a]	ovoconf Variable in Audit Name Space
	notification schedule		SCHEDULE
HPOM Other	User modifies a notification schedule	MINOR	OM_CFG_CHG_NOTIFY_SCHEDULE
HPOM Other	User deletes a notification schedule	MINOR	OM_CFG_CHG_NOTIFY_SCHEDULE
HPOM Other	User creates a notification service	MINOR	OM_CFG_CHG_NOTIFY_SERVICE
HPOM Other	User modifies a notification service	MINOR	OM_CFG_CHG_NOTIFY_SERVICE
HPOM Other	User deletes a notification service	MINOR	OM_CFG_CHG_NOTIFY_SERVICE
HPOM Other	User creates a node group	MINOR	OM_CFG_ADD_NODE_GRP
HPOM Other	User modifies a node group	MINOR	OM_CFG_ADD_NODE_GRP
HPOM Other	User deletes a node group	MAJOR	OM_CFG_ADD_NODE_GRP
HPOM Other	User changes the MSI setting - Enable	SERIOUS	OM_CFG_CHG_MSI
HPOM Other	User changes the MSI setting - Disable	SERIOUS	OM_CFG_CHG_MSI
HPOM Other	User changes the MSI setting - Allowing for externally defined actions	SERIOUS	OM_CFG_CHG_MSI
HPOM Other	User assigns a category to a policy	MINOR	OM_CFG_CHG_POLICY

[a]Minor, Major, Serious, or Internal

HPOM-Object Audit Areas , continued

Object Audit Area	Use Case	Default Audit Level [a]	ovoconf Variable in Audit Name Space
HPOM Other	User deassigns a category from a policy	MINOR	OM_CFG_CHG_POLICY
HPOM Other	User assigns a category to a policy group	MINOR	OM_CFG_CHG_POLICY_GRP
HPOM Other	User deassigns a category from a policy group	MINOR	OM_CFG_CHG_POLICY_GRP
HPOM Other	User creates a category directory under the instrumentation directory	MINOR	OM_CFG_ADD_CATEGORY
HPOM Other	User removes a category directory under the instrumentation directory	MINOR	OM_CFG_DEL_CATEGORY
HPOM Other	User creates a policy group	MINOR	OM_CFG_ADD_POLICY_GRP
HPOM Other	User assigns a node to a node group	MINOR	OM_CFG_CHG_NODE_GRP
HPOM Other	User deassigns a node from a node group	MINOR	OM_CFG_CHG_NODE_GRP
HPOM Other	User creates a new policy	MINOR	OM_CFG_ADD_POLICY
HPOM Other	User modifies a policy	MINOR	OM_CFG_CHG_POLICY
HPOM Other	User deletes a policy	MAJOR	OM_CFG_DEL_POLICY

[a]Minor, Major, Serious, or Internal

HPOM-Object Audit Areas , continued

Object Audit Area	Use Case	Default Audit Level [a]	ovoconf Variable in Audit Name Space
HPOM Other	User edits a policy using the `poledit` application	MINOR	`OM_CFG_CHG_POLICY`

HPOM Scripts and Binaries

Table 74 lists the HPOM processes that you can select for auditing

Table 74: HPOM Scripts and Binaries Audit Areas

Audit Area	Use Case	Default Audit Level [b]	ovoconf Variable in Audit Name Space
HPOM script/binaries access - Execute	Command-line interface (CLI) starts	MINOR	none
HPOM script/binaries access - Execute	Scheduled action runs	MAJOR	`OM_AGT_RUN_SCHED_ACT`
HPOM script/binaries access - Execute	License check fails when adding a new node	MAJOR	`OM_LICENSE_CHECK_FAILURE`
HPOM script/binaries access - Execute	License check fails when adding a new agent-less node	MAJOR	`OM_LICENSE_CHECK_FAILURE`
HPOM script/binaries access - Execute	Nightly license check fails	MAJOR	`OM_LICENSE_CHECK_FAILURE`

HPOM Processes

Table 75 lists the HPOM processes that you can select for auditing.

[a]Minor, Major, Serious, or Internal

[b]Minor, Major, Serious, or Internal

Table 75: HPOM Process Audit Areas

Audit Area	Use Case	Default Audit Level [a]	ovoconf Variable in Audit Name Space
HPOM Startup	Administrator starts the HP Operations agent software locally	MINOR	OM_AGT_START
HPOM Startup	Administrator starts the HP Operations agent software remotely	MINOR	OM_AGT_START
HPOM Startup	Administrator starts the HP Operations management-server software	MAJOR	OM_SV_START
HPOM Shutdown	Administrator shuts down the HP Operations management-server software	SERIOUS	OM_SV_STOP
HPOM Shutdown	Administrator shuts down the HP Operations agent software locally	SERIOUS	OM_AGT_STOP_ON_SV
HPOM Shutdown	Administrator shuts down the HP Operations agent software remotely	MAJOR	OM_AGT_STOP

[a]Minor, Major, Serious, or Internal

Appendix D: Manual Pages

In this Appendix

This appendix describes how to access the manual pages that are available for HP Operations Manager (HPOM) and HP Service Navigator (Service Navigator) and lists all the manual pages that are available for the different areas. The information in this appendix covers the following areas:

Access to HPOM Manual Pages

You can access the HPOM manual pages from the command line, from online help, or in HTML format on your management server.

Accessing Manual Pages from the Command Line

To access an HPOM manual page from the command line, enter the following command:

```
# man <manual_page-page_name>
```

Printing Manual Pages from the Command Line

To print an HPOM manual page from the command line, enter the following command:

```
# man <manual_page-page_name> | col -lb | lp -d printer_name
```

Accessing Manual Pages in HTML Format

To access the HPOM manual pages in HTML format, for example, from an Internet browser, use the following URL in the browser:

- Standard connection:

  http://*<management_server>*:8081/ITO_MAN
- Secure connection:

  https://*<management_server>*:8444/ITO_MAN

In this instance, *<management_server>* is the fully qualified hostname of your HPOM management server.

HPOM Manual Pages

This section describes the manual pages that are available in HPOM.

Table 76: HPOM Manual Pages

Manual Page	Description
call_sqlplus.sh(1)	Calls SQL*Plus (Oracle) or psql (PostgreSQL).
inst.sh(1m)	Installs HPOM software on managed nodes.
inst_debug(5)	Debugs an installation of the HPOM agent software.
ito_op(1m)	Launches the HPOM Java GUI or Service Navigator GUI.
ito_op_api_cli(1m)	Enables calls to the Java GUI Remote APIs.
opcbackup_offline(1m)	*Oracle only:* Interactively backs up the HPOM environment for the database.
opcbackup_offline(5)	Backs up the HPOM configuration.
opcdelmsg(1m)	Removes messages from the message-manager queue even while management-server processes are running. Only messages matching all specified criteria are deleted.
opcrecover_offline(1m)	*Oracle only:* Interactively restores the HPOM environment for the database.
opcrecover_offline(5)	Restores the HPOM configuration.
opcack(1m)	Externally acknowledges active messages.
opcunack(1m)	Externally unacknowledges history messages of a selected operator.
opcackmsg(1m)	Externally acknowledges active messages using message IDs.
opcackmsgs(1m)	Externally acknowledges active messages using specific message attributes.
opcactivate(1m)	Activates a pre-installed HPOM agent.

HPOM Manual Pages , continued

Manual Page	Description
opcadddbf(1m)	*Oracle only:* Adds a new datafile to an Oracle tablespace.
opcagt(1m)	Manages agent processes on a managed node.
opcappl(1m)	Manages applications and application groups including assigning applications or application groups to (and deassigning them from) other application groups.
opcagtutil(1m)	Parses the agent-platform file and performs operations with extracted data.
opccfgdwn(1m)	Downloads configuration data from the database to flat files.
opccfgout(1m)	Configures condition status variables for scheduled outages in HPOM.
opccfgupld(1m)	Uploads configuration data from flat files to the database.
opccfguser(1m)	Configures HPOM on UNIX operators and is used for assigning user profiles, deassigning user profiles, and configuring the user-responsibility matrix.
opcconfig(1m)	Configures an HPOM management server.
opccsa(1m)	Provides the functionality for listing, mapping, granting, denying, and deleting specified certificate requests.
opccsacm(1m)	Performs ovcm functionality, for example, manually issuing new node certificate or using the installation key.
opcdbidx(1m)	*Oracle only:* Upgrades the structure of the HPOM database.
opcdbinit(1m)	Initializes the database with the default configuration.
opcdbinst(1m)	Creates or removes the HPOM database schema.
opcdbpwd(1m)	*Oracle only:* Changes the password of the HPOM database user opc_op.
opcdbsetup(1m)	Creates the HPOM database.
opcdcode(1m)	Views HPOM encrypted policy files.
opcdistchk	Checks policies and instrumentation on a managed node and compares them with the ones indicated in the server inventory.
opcerr(1m)	Displays instruction text for HPOM error messages.
opcgetmsgids (1m)	Lists the IDs associated with an original message ID (provided as the argument with the command), for example: the ID of correlated messages, messages

HPOM Manual Pages , continued

Manual Page	Description
	modified by an external MSI, and so on.
opchbp(1m)	Switches heartbeat polling of managed nodes on or off.
opchistdwn(1m)	Downloads HPOM history messages to a file.
opchistupl(1m)	Uploads history messages to the HPOM database.
opcinstrumcfg (1m)	Manages category information in the file system and database level simultaneously.
opcinstrumdwn (1m)	Downloads all instrumentation files to a single flat file that can then be manually deployed to an HPOM agent.
opcmack(1)	Acknowledges an HPOM message by specifying the message ID.
opcmom(4)	Provides an overview of HPOM flexible-management functionality.
opcmomchk(1)	Checks the syntax of HPOM flexible-management policies.
opcmon(1)	Forwards the value of a monitored object to the HPOM monitoring agent on the local managed node.
opcmsg(1)	Submits a message to the HPOM message interface.
opcnode(1m)	Maintains nodes, node groups and policy assignments in HPOM.
opcownmsg(1m)	Sets, unsets, and changes HPOM message ownership.
opcmsgchg(1m)	Modifies the severity and text of an HPOM message.
opcpat(1)	Tests a program for HPOM pattern matching.
opcragt(1m)	Remotely manages agent services for HPOM on a managed node.
opcskm(3)	Manages secret keys.
opcsqlnetconf (1m)	*Oracle only:* Configures the HPOM database to use an Oracle Net connection.
opcsv(1m)	Manages HPOM management-server processes.
opcsw(1m)	Sets the software status flag in the HPOM database.
ovswitchuser (1m)	Switches ownership of the HPOM agents.

HPOM Manual Pages , continued

Manual Page	Description
opcpolicy(1m)	Maintains policies in files; replaces opctempl(1m).
opcpolicy(1m)	Enables and disables policies.
opctmpldwn(1m)	Downloads and encrypts HPOM message-source policies.
opcwall(1)	Sends a message to all HPOM users who are currently logged in.
ovocomposer (1m)	Performs tasks related to HP Composer.
ovocomposer(5)	Describes the Correlation Composer, an HPOM event-correlation feature.
ovtrap2opc(1m)	Converts the trapd.conf file and the HPOM policy file.

Manual Pages for the HPOM API

This section describes manual pages that are available for HPOM application program interfaces (APIs).

Table 77: HPOM API Manual Pages

Manual Page	Description
opcmon(3)	Forwards the value of a monitored object to the HPOM monitoring agent on the local managed node.
opcmsg(3)	Submits a message to HPOM.

Manual Pages for Service Navigator

This section describes manual pages for the Service Navigator.

Table 78: Service Navigator Manual Pages

Manual Page	Description
opcservice(1m)	Configures Service Navigator.
opcsvcattr(1m)	Adds, changes, or removes service attributes.
opcsvcdwn(1m)	Downloads status logs of Service Navigator to a file.
opcsvcterm(1m)	Emulates an interface to Service Navigator. The interface inputs Extensible

Service Navigator Manual Pages, continued

Manual Page	Description
	Markup Language (XML) markup into `stdin` and outputs Extensible Markup Language (XML) markup to `stdout`.
`opcsvcupl(1m)`	Uploads the Service Navigator service-status logs to the HPOM database.

Appendix E: Automatic Service Actions

About Automatic Service Actions

HP Operations Manager Java GUI offers a possibility to perform a scope of service actions predefined for the particular services in the service configuration file. These service actions are always triggered by the Java GUI operator, and are used for faster navigation in the Java GUI. See "Defining Automatic Service Actions" on page 793 for more information about defining these actions.

In addition to these predefined service actions, you can also configure **automatic** service actions which are performed when the service status changes, for example, if the service severity changes to critical. These actions can be associated with each of the possible severity levels, and are defined as commands executed on the HPOM management server, see "Defining Automatic Service Actions" on page 793 for more information.

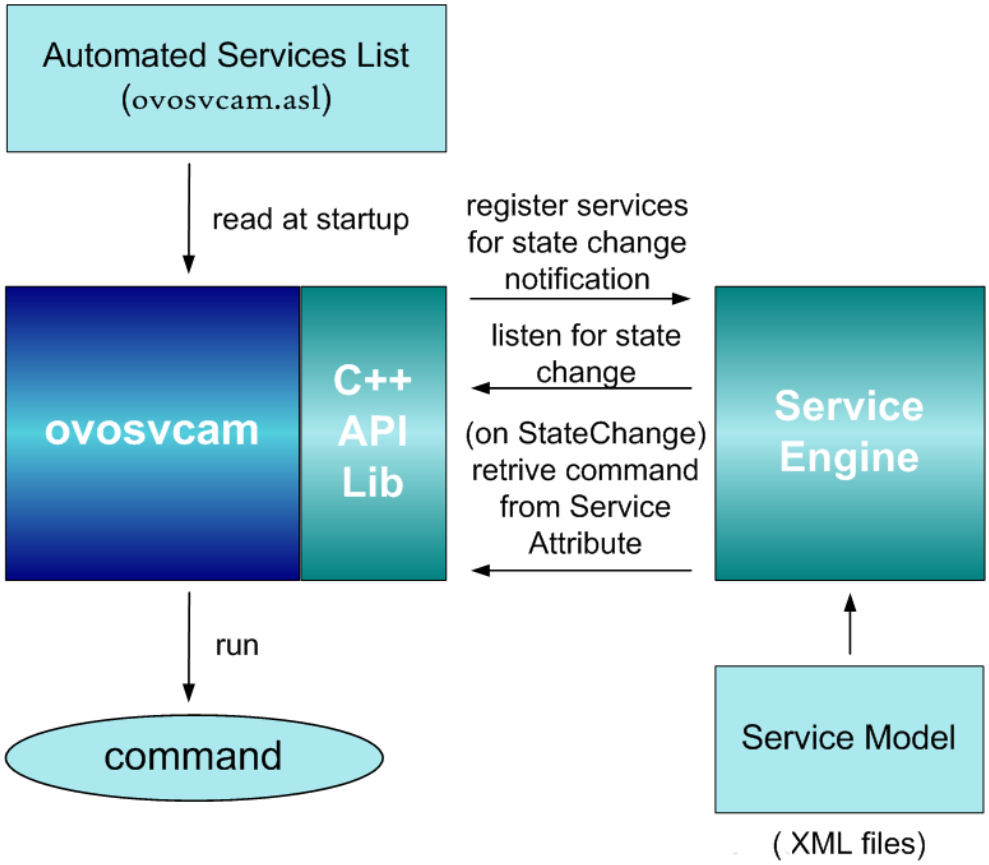
Services, to which automatic action(s) are associated with, are referred to as **automated** in the remainder of the document.

For details on how to implement these kind of service actions, see "How Automatic Service Actions Work" on page 788 .

How Automatic Service Actions Work

Automatic service actions are based on the **HPOM Service Navigator Action Manager (opcsvcam)** utility that communicates with the HPOM service engine (see Figure 27 for graphical presentation of service engine interfaces).

Figure 27: Service Engine Interfaces

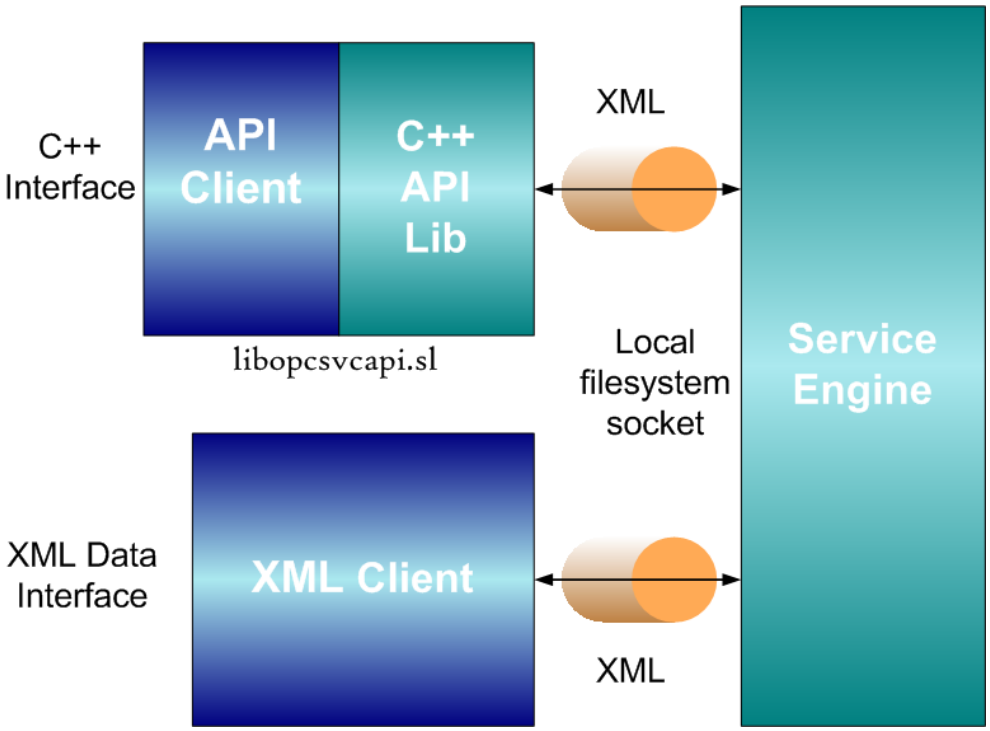


The `opcsvcam` is a service engine listener program, designed using C++ and the service engine APIs (see Figure 28 for the presentation of the `opcsvcam` design).

Example API programs are available on the HPOM management server at the following location:

`/opt/OV/OpC/examples/progs/svcapi`

Figure 28: opcsvcam Design



The `opcsvcam` runs continually on the HPOM management server, listens for status changes of the services listed in the automated services list (see "Automated Services List" on page 792 for more information), and triggers the appropriate automatic action upon the specified status change.

Specifying automatic actions and associating them with the status changes is detailed in the "Defining Automatic Service Actions" on page 793.

Automatic Actions Configuration Files Locations

The `opcsvcam` utility and some examples of configuring automatic service actions are placed on the HPOM management server, at the following locations:

Filename	Description
`opcsvcam`	Executable binary Service Navigator automatic action, located at `/opt/OV/bin/OpC/`
`email_svcam.xml`	Example service definition file with defined actions, located at `/opt/OV/OpC/examples/services/`
`opcsvcam.asl`	Example configuration file - configures services with defined actions in `email_svcam.xml`, located at `/opt/OV/OpC/examples/services/`

Enabling Automatic Actions

Before Enabling Automatic Actions

Before you start with enabling automatic actions, you should determine the following:

- **Automated services subset**

  Decide for which services the automatic service actions will be performed. Clear up which services would require automatic actions the most, do not include each service in the service hierarchy. Consult the "Best Practices and Recommendations" on page 791 before making the decision.
- **Automatic action details**

  For each automated service, decide which severity level will trigger the automatic action, and define the command which will be executed.

To Enable Automatic Actions

1. **Create automated services list**.

   Include services that you have chosen for monitoring in the automated services list. For details about this list, see "Automated Services List" on page 792.
2. **Define automatic actions for each automated service.**

   Automatic actions should be defined as commands that are executed on severity change to a specified level for each automated service. For more information, see "Defining Automatic Service Actions" on page 793.
3. **Activate/upload the modified Service Navigator configuration.**

   For more information, see "Activating the Service Configuration" on page 345.

Best Practices and Recommendations

Follow the best practices and recommendations listed below if you plan to set automatic service actions in your Service Navigator environment:

- It is not appropriate to set the automatic service actions for each service in the service hierarchy. For example, a message with severity critical would result in changing the severity level for a number of services. If you set the automatic service action to, for example, '`Send a notification`' for each service which severity status changes to critical, this could trigger too many notifications for just one event.

  Also, setting automatic actions for each severity level would rather result in confusion than in enhanced monitoring of services. It would be sufficient to set automatic service actions for the severity critical and/or major.

- Identify which services would require automatic actions upon service state changes, some good examples are the following:
  - Application service

    Example of an action: send an e-mail to the application owner
  - LOB service

    Example of an action: send an e-mail or a report to the LOB owner
  - Database service

    Example of an action: notify the Database Administrator

Automated Services List

Services with their severity status monitored for automatic actions (automated services) are organized in an **automated services list (opcsvcam.asl)**, which is read at HPOM startup by the `opcsvcam` utility (see "How Automatic Service Actions Work" on page 788 to learn more about the `opcsvcam`).

Automated services list is a simple ASCII file, placed on the HPOM management server at the following location:

`/etc/opt/OV/share/conf/OpC/mgmt_sv/`

Note: Each automated service name requires one line in a list. Make sure you specify the service name, not the label.

The following is an example of the `opcsvcam.asl` file where two services, `email` and `america`, are associated with the automatic service actions as described in the "Defining Automatic Service Actions" on page 793:

Automated Services List

```
# File:         opcsvcam.asl
# Last Update: 27-March-2006
#
# This file contains a list of service names of services that will be
# monitored by the HPOM Service Navigator Action Manager (opcsvcam).  When the
status
# of any of these services changes, opcsvcam will execute an auto-action
# command (if defined).
#
# Service names follow below.
email
america
```

Defining Automatic Service Actions

Automatic service actions are defined with special service attributes on the automated services. These attributes contain the following parameters:

Parameter	Description
<name>	Associated with a severity which, when reached by a service, triggers the automatic action. It can be one of the following: `SevNormal`, `SevWarning`, `SevMinor`, `SevMajor` and `SevCritical`.
<value>	Automatic action (command).

The following are some examples of commands executed as automatic actions:

- **To send an e-mail:**

  ```
  echo “Subject: Database Svc Alert\nSAP Database is in Critical State” | sendmail dba@xyzcorp.com
  ```

- **To create a trouble-ticket in help-desk system:**

  ```
  sd_event –f config.sd_event –v event_id=1234 description=“SAP Database has changed to CRITICAL state” information=“Operations is working the issue…”
  ```

- **To forward a message to the target server to update the service hierarchy:**

  ```
  opcmsg a=opcsvcam o=database_ins msg_grp=“SFM” msg_t=“StateChange on service” severity=major
  ```

To learn how to define automatic service actions in Service Navigator, see "Defining Actions in Service Navigator" on page 793.

Automatic Actions Parameters

The following parameters forward service related information to the action:

Parameter	Description
`$OLDSEVERITY`	Resolves into old severity.
`$NEWSEVERITY`	Resolves into a new severity.
`$SERVICENAME`	Resolves into a service name.

Defining Actions in Service Navigator

Automatic service actions can be defined in Service Navigator in the service configuration file. For example, for the `opcsvcam.asl` file presented in the "Automated Services List" on page 792, the

corresponding service configuration file could contain actions defined in the following example. This example is an excerpt from the `email_svcam.xml` example file provided with the installation. For a list of all installed files and their location, see "Automatic Actions Configuration Files Locations" on page 790.

In the example, the `<Attribute>` tag is used for defining the following automatic service action: When the severity of service `america` changes to critical (*`<Name>`* tag: `SevCritical`), the following HPOM message is sent: `Severity on service america changed to CRITICAL` (*`<Value>`* tag: `opcmsg a=a o=opcsvcam msg_t="Severity on service america changed to CRITICAL"`).

Likewise, similar HPOM messages are sent when the severity of the service `email` reaches values - major or critical.

To learn more about the service configuration file and its syntax, see the "Service Configuration File" on page 364.

Defining Automatic Service Actions in the Service Configuration File

```
<?xml version="1.0"?>
<Services xmlns="http://www.hp.com/OV/opcsvc"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xsi:schemaLocation="http://www.hp.com/OV/opcsvc
/etc/opt/OV/share/conf/OpC/mgmt_sv/dtds/service.xsd">

    <Service>
      <Name>america</Name>
      <Label>america</Label>
      <CalcRuleRef>america_1</CalcRuleRef>
<Attribute>
   <Name>SevCritical</Name>
   <Value>opcmsg a=a o=opcsvcam msg_t="Severity on service america
changed to CRITICAL"</Value>
</Attribute>
      <Source>
        <Composition/>
        <ServiceRef>email_node1</ServiceRef>
      </Source>
      <Source>
        <Composition/>
        <ServiceRef>email_node2</ServiceRef>
      </Source>
   </Service>
   <Service>
      <Name>email</Name>
      <Label>E-Mail</Label>
<Attribute>
        <Name>SevMajor</Name>
   <Value>opcmsg a=a o=opcsvcam msg_t="Severity on service email
changed to MAJOR"</Value>
      </Attribute>
<Attribute>
   <Name>SevCritical</Name>
        <Value>opcmsg a=a o=opcsvcam msg_t="Severity on service email
changed to CRITICAL"</Value>
</Attribute>
      <Source>
        <Composition/>
        <ServiceRef>america</ServiceRef>
      </Source>
      <Source>
        <Composition/>
        <ServiceRef>europe</ServiceRef>
      </Source>
     </Service>

</Services>
```

Send Documentation Feedback

If you have comments about this document, you can contact the documentation team by email. If an email client is configured on this system, click the link above and an email window opens with the following information in the subject line:

Feedback on Administrator's Reference Guide (Operations Manager 9.22)

Just add your feedback to the email and click send.

If no email client is available, copy the information above to a new message in a web mail client, and send your feedback to docfeedback@hpe.com.

We appreciate your feedback!


Java™
COMPATIBLE