HP OpenView Performance Agent

For the Sun Solaris Operating System

Software Version: 4.60

Installation and Configuration Guide

Manufacturing Part Number: B7491-90079
Document Release Date: December 2006
Software Release Date: December 2006

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 Notices

Trademark Notices

UNIX® is a registered trademark of The Open Group.

Adobe® and Acrobat® are trademarks of Adobe Systems Incorporated.

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

Microsoft® is a U.S. registered trademark of Microsoft Corporation.

All other product names are the property of their respective trademark or service mark holders and are hereby acknowledged.

Support

You can visit the HP OpenView Support web site at:

www.hp.com/managementsoftware/support

HP OpenView online support provides an efficient way to access interactive technical support tools. As a valued support customer, you can benefit by using the support 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 find more information about access levels, go to:

www.hp.com/managementsoftware/access_level

To register for an HP Passport ID, go to:

www.managementsoftware.hp.com/passport-registration.html

Contents

1 Installing or Upgrading OpenView Performance Agent

Introducing OV Performance Agent

HP OpenView Performance Agent (OVPA) captures performance, resource, and transaction data from your Solaris system. Using minimal system resources, the software continuously collects, summarizes, time stamps, and detects alarm conditions in current and historical resource data across your system. You can analyze the data using spreadsheet programs, Hewlett-Packard analysis products such as OV Performance Manager (OVPM), or third-party analysis products. Also, OV Performance Agent provides data access to OV Performance Manager and sends alarm notifications to HP OpenView Network Node Manager (NNM) and OpenView Operations (OVO).

OV Performance Manager (OVPM) in this document refers only to versions 4.0 and later. The name OVPM 3.x is used throughout this document to refer to the product that was formerly known as PerfView.

OV Performance Agent uses data source integration (DSI) technology to receive, alarm on, and log data from external data sources such as applications, databases, networks, and other operating systems.

The comprehensive data logged and stored by OV Performance Agent allows you to:

- Characterize the workloads in the environment.
- Analyze resource usage and load balance.
- Perform trend analyses on historical data to isolate and identify bottlenecks.
- Respond to error conditions.
- Perform service-level management based on transaction response time.
- Perform capacity planning.

- Solve system management problems before they arise.

For a comprehensive description of OV Performance Agent, see the *HP OpenView Performance Agent for UNIX User's Manual*.

Installation Requirements

Before installing OV Performance Agent, make sure that your system meets the requirements described in this section. Certain system and configuration prerequisites are necessary for OV Performance Agent to operate properly on your system.

Hardware

OV Performance Agent generally runs on hardware platforms supporting the operating system, including:

- Sun 4c
- Sun 4d
- Sun 4m
- Sun 4n
- Sun 4u
- E 10000, and Sunfire

Software

- This version of OV Performance Agent requires the Sun Solaris 7 operating environments or later. It is recommended to have at least the "End User System Support" level of operating system installed on your Sun system. If the "Core System Support" level of operating system is installed on your system, you have to additionally install the following two operating system packages.
- SUNWlibC
- SUNWlibms

  You can determine which packages are installed by using the `pkginfo` command:

  `pkginfo SUNWlibC SUNWlibms`

  If the command returns:

```
ERROR: information for "SUNWlibC" was not found
system      SUNWlibms      Sun Workshop Bundled shared libm
```

You have to install the appropriate package (SUNWlibC). The packages can be found on the Sun Solaris distribution media.

Communication Protocols

OV Performance Agent supports the following communication protocols:

- HTTP(S) 1.1
- NCS 1.5.1
- DASCOM DCE 1.1.4.15.3 for Sun Solaris 7 and later (`HPlwdce`, the lightweight DCE client bundled within OV Performance Agent)
- IBM DCE 3.1 for Sun Solaris 7 and 8
- IBM DCE 3.2 for Sun Solaris 9

  If you intend to use a commercial DCE product, you must have `HPlwdce` package installed as well. See Installing OV Performance Agent and Keeping the Installed DCE on page 18.

Disk Space

OV Performance Agent installs in the `/opt/OV/ and /opt/perf/` directories and creates its log and status files in the `/var/opt/OV/ and /var/opt/perf/` directories.

- For first time installation of OV Performance Agent, 70 MB of disk space is required in the `/opt/OV/ and /opt/perf/` directories.
- For OVPA databases and status files, allow for 125 MB of disk space in the `/var/opt/OV/ and /var/opt/perf/` directories.

If you do *not* have enough space in your `/opt/perf` and/or `/var/opt/perf` directories, you must install OV Performance Agent on symbolic links. Refer to Installing OV Performance Agent on Symbolic Links on page 14.

For a description of how the `parm` file is used to limit and configure log file data storage, see the “parm File” section in Chapter 2 of your *HP OpenView Performance Agent for UNIX User’s Manual*.

Install or Upgrade Procedures

OV Performance Agent comes on a CD installation media. The size of the product is approximately 70 MB, including the product documentation.

If you have previously installed version of OVPA or GlancePlus on the system, stop any performance tools or processes that may be running. For instructions, refer to the section, Stopping Active Performance Tools or Processes. For installation instructions, refer to the section, Installing OV Performance Agent.

Stopping Active Performance Tools or Processes

1. Log in as user **root**.
2. Run `perfstat` to check for active performance tools by typing:

   /opt/perf/bin/perfstat

   If `perfstat` reports any active performance tools such as GlancePlus, stop them. (Make sure that users have exited these tools before doing so.)
3. If a previously-installed version of OV Performance Agent is running, you must stop it by typing:

   /opt/perf/bin/mwa stop

Customized configuration files such as the `parm`, `alarmdef`, `ttd.conf` and `perflbd.rc` as well as any customized log files will *not* be overwritten by the new installation. The new configuration files are installed in the `/opt/perf/newconfig` directory.

4. If you stop `ttd`, any ARM-instrumented applications that are running *must* also be stopped before you restart `ttd` and OV Performance Agent processes.Run `perfstat` again to ensure that no performance tools or processes are active. When all tools or processes have been stopped, proceed with the installation.If you are using the HP lightweight DCE bundled with OV Performance Agent and you do *not* have a commercial DCE installed on your system, remove the `/opt/dcelocal` directory if it exists. Otherwise the `HPlwdce` lightweight DCE may *not* install properly.

Installing OV Performance Agent

While installing OV Performance Agent, you can specify the data communication protocol to be used. If you are installing OV Performance Agent for the first time, by default, the data communication protocol is set to HTTP. If you are upgrading OV Performance Agent to the current version, the previously used protocol is retained. For a detailed description of `install.ovpa` options, see The install.ovpa Script on page 16.

If you have HP OpenView Operations (OVO) installed on your system, see Installing OV Performance Agent with OV Operations Agent Installed on Your System on page 15.

The HTTP communication protocol is always enabled, irrespective of the protocol you have selected during installation. The daemons used for HTTP data communication are always installed and active on your system.

The table below lists the data communication protocol used with different installation options of OV Performance Agent.

Table 1 OV Performance Agent installation options

Options	OVPA Standalone (No OVO 7.x)	OVPA on OVO 7.x installation in DCE mode	OVPA on OVO 7.x installation in NCS mode
Protocol			
`-p http`	HTTP	HTTP	HTTP
`-p dce`	DCE	DCE	NCS
`-p ncs`	NCS	DCE	NCS
No protocol specified	HTTP in a first time installation of OVPA and the previously used protocol on upgrade to the current version of OVPA		

To install:

1. Make sure you are logged in as user **`root`**.
2. Insert your installation CD-ROM into the drive. The CD-ROM is automatically mounted on Sun Solaris systems.

3 Change to the CD-ROM directory by typing:

cd /<*directory*>

where <*directory*> is your CD-ROM directory.

4 Type **ls** to verify that you are in the correct directory. You will see the `install.ovpa` script, the `readme.ovpa` file, and the `paperdocs` and `rpmtools` directory listed on the screen.

5 Run the install script.

To install using the HTTP communication protocol, type:

./install.ovpa -p http

To install using the DCE communication protocol, type:

./install.ovpa -p dce

To install using the NCS communication protocol, type:

./install.ovpa -p ncs

The `install.ovpa` script checks for OV Operations 7.x agent installed on your system as described in Installing OV Performance Agent with OV Operations Agent Installed on Your System on page 15.

If you specify `./install.ovpa -p ncs`, OV Performance Agent will be started using `dced` that emulates the NCS local location broker daemon (`llbd`). In order to run OV Performance Agent in genuine NCS mode using `llbd`, you must manually edit the `/etc/default/ovpa` file and set the `MWA_LLBD_COMMAND` variable to `/opt/perf/bin/llbd`. See The /etc/default/ovpa File on page 27 in Chapter 2 for more information.

For details on changing the communication protocol after installation, see Changing Protocols on page 26.

The `install.ovpa` script automatically starts all OV Performance Agent processes in its configured communication mode. If you do *not* want OV Performance Agent to start automatically, run the install script with the option `-R`. See The The install.ovpa Script on page 16 for a detailed description of `install.ovpa` options. The OV Performance Agent processes are also started or stopped automatically if you reboot or shutdown. See Starting and Stopping OV Performance Agent on page 22.

6 Exit the CD-ROM directory by typing:

cd /

7 You can unmount the CD-ROM by typing:

```
eject
```

OV Performance Agent installation is now complete. Go to Chapter 2, Starting and Running OV Performance Agent, for details on other tasks you need to perform to get OV Performance Agent up and running.

Installing OV Performance Agent on Symbolic Links

If you do *not* have enough space in `/opt/OV/`, `/opt/perf`, `/var/opt/perf`, or `/var/opt/OV/` directories, select an alternative directory (or directories) and symbolically link `/opt/OV/`, `/opt/perf`, `/var/opt/perf`, or `/var/opt/OV/` to these directories.

For example:

`ln -s /`*<dir>*`/var/opt/perf /var/opt/perf`

and/or

`ln -s /`*<dir>*`/opt/perf /opt/perf`

where <*dir*> stands for directory of your choice.During the installation process, the `install.ovpa` script finds symbolic link(s) and will continue with the installation, showing on-screen message(s):

```
NOTE: Found symbolic link for /opt/perf -> /<dir>/opt/perf
Installation will continue on this symbolic link
```

and/or

```
NOTE: Found symbolic link for /var/opt/perf -> /<dir>/var/opt/
perf
Installation will continue on this symbolic link
```

For more information and late-breaking news about OV Performance Agent, see your Release Notes. You can find them in `/opt/perf/ReleaseNotes/OVPA`

If you are also running the Glance product on your system, be sure to update Glance to the same release version as OV Performance Agent. Both OV Performance Agent and Glance must always be the same version.

Installing OV Performance Agent with OV Operations Agent Installed on Your System

If you are installing OV Performance agent for the first time, and if no communication protocol is specified, the default data communication mode is HTTP.

When `install.ovpa` is executed, the installation script automatically detects if the OV Operations 7.x agent is installed on your system and which communication protocol it is using.

While upgrading OV Performance Agent, if OV Operations 7.x agent is detected on your system, the protocol specified using the `-p` option is ignored, and OV Performance Agent is started using the same protocol as the OV Operations agent. If OV Operations 7.x agent is detected on your system:

- If NCS service is used, the `install.ovpa` script is started with the `-p ncs` option, enforcing the deployment of the NCS communication protocol.
- If DCE service is used, the `install.ovpa` script is started with the `-p dce` option, enforcing the deployment of the DCE communication protocol.

See The install.ovpa Script on page 16 for a detailed description of `install.ovpa` options.

The presence of OV Operations 8.x agent on your system does not affect the default installation behavior of OV Performance Agent. During first time installation of OV Performance Agent on systems that have OV Operations 8.x agent installed, OV Performance Agent is started in the HTTP mode. If you are upgrading, OV Performance Agent is started using the same communication protocol as the previous installation. For more information on how OV Performance Agent is installed, see Installing OV Performance Agent on page 12.

Deploying OV Performance Agent Using OV Operations

If you are using HP OpenView Operations for UNIX 7.x or 8.x, you can install OV Performance Agent from the management server to a Sun Solaris managed node.

For installation instructions from an HP OpenView Operations for UNIX 7.x management server, refer to the chapter “About OpenView Performance Agent for HP-UX and Sun Solaris” in the *HP OpenView VantagePoint Operations for UNIX Administrator's Reference, Volume II*.

For installation instructions from an HP OpenView Operations for UNIX 8.x management server, refer to the chapter “HP OpenView Performance Agent” in the *HP OpenView Operations for UNIX Administrator's Reference*.

The install.ovpa Script

To install OV Performance Agent, you must run the `install.ovpa` script. This section describes the installation script command line options, which can be used for more advanced installations. The syntax of the command is as follows:

`install.ovpa [-hnR] [-p dce | ncs | http] [-a admin] [-d spool_dir]`

The command line options have the following meaning:

`-a admin`	Use `admin` file for the `pkgadd`. By default the script creates and uses its own `admin` file. See *admin(4)* for more details.
`-d spool`	Use `spool` directory for OV Performance Agent packages location. Default is `./rpmtools`.
`-h`	Display this message and exit.

`-n` Do *not* install. Only display the installation parameters and exit.

`-R` Do *not* start OV Performance Agent upon successful installation. By default, OV Performance Agent is automatically started.

`-p proto` Enforce the deployment of the selected communication protocol. The `proto` can be one of the following:

- `http` Use HTTP communication protocol
- `dce` Use DCE communication protocol
- `ncs` Use NCS communication protocol.

By default, the installation procedure examines the system configuration and determines which protocol to use. See Installing OV Performance Agent with OV Operations Agent Installed on Your System.

All required product packages are installed by default.

Switching from the HP Lightweight DCE to a Commercial DCE

If you are using the HP lightweight DCE package bundled with OV Performance Agent and you decide to install one of the commercial DCE products, you must:

- Remove your currently installed OV Performance Agent.
- Install the commercial DCE product.
- Re-install OV Performance Agent, using the `-p dce` option.

Switching from a Commercial DCE to the HP Lightweight DCE

The HP lightweight DCE gets installed by default in a OV Performance Agent installation.

If you are using a commercial DCE and want to use the HP lightweight DCE instead, you will have to:

- Remove your currently installed OV Performance Agent.

- Remove your commercial DCE.
- Re-install OV Performance Agent, using the `-p dce` option.

Installing OV Performance Agent and Keeping the Installed DCE

If you have a commercial DCE product (IBM DCE 3.1 for Sun Solaris 7) installed and running on your system, you must have HPlwdce package installed as well. If you try to install OV Performance Agent without the HPlwdce package, the installation will fail with the warning message:

The <HPlwdce> package “Lightweight DCE client” is a prerequisite and should be installed.

When you perform installation with HPlwdce package, the installation script detects a DCE client already exists, and the `dce` daemon will not be started. The commercial DCE will be deployed for communication with other Performance products.

Removing OV Performance Agent

If you need to remove OV Performance Agent from a system, use the `ovpa.remove` script that is in the `/opt/perf/bin/` directory. However, before removing OV Performance Agent, make sure you archive any log files that were created. These files contain performance data for that system and can be used to extract or view data at a later time.

During the removal process, you will be asked if you want to remove the OV Performance Agent configuration and logfiles:

`"Do you want to remove OVPA configuration and logfiles in the /var/opt/perf/datafiles and /var/opt/perf directory?"`

Answer **`N`** (no) if you want to keep the configuration and log files at the original location.

Note that these files will *not* be overwritten by a new OV Performance Agent installation. The new configuration files are uploaded to the `/opt/perf/newconfig` directory.

The script supports the following command line options for a more flexible product removal. The syntax of the command is as follows:

`ovpa.remove [-fhn] [-a admin]`

`-f`	Force removal. Do not prompt for confirmation before removing the product (non-interactive mode).
`-a admin`	Use `admin` file for the `pkgrm`. By default the script creates and uses its own `admin` file. See *admin(4)* for more details.
`-h`	Help: Displays this usage message.
`-n`	Do not remove. Only display the removal parameters and exit.

It is possible that some product packages may remain installed on the system, if those packages are shared across other OpenView products and are required by other tools. They will be removed only when the last tool requiring them is also removed.

2 Starting and Running OV Performance Agent

Introduction

This chapter describes the tasks involved in starting up and running OV Performance Agent after it has been installed on your Solaris system. The following topics are discussed:

- Starting and Stopping OV Performance Agent
- Communicating Across Firewall
- Configuring Secure Communication
- Configuring Data Sources
- Defining Alarms

If you are planning to log data from other sources using data source integration (DSI), and have *not* yet done so, read the *HP OpenView Performance Agent for UNIX Data Source Integration Guide*.

Starting and Stopping OV Performance Agent

When installation is complete, you can start OV Performance Agent. The OV Performance Agent scripts, `ovpa` and `mwa`, let you start all or some processes and stop or restart currently running processes.

If you are installing OV Performance Agent for the first time, the default data communication mode is HTTP. If you are upgrading OV Performance Agent, the previously used DCE or NCS data communication mode is enabled by default. For information on changing the data communication protocol, see Changing Protocols on page 26.

Depending on the data communication protocol you want to enable, you can use the `ovpa` or `mwa` script to start or stop OVPA.

It is recommended that you use the `ovpa` script to start OVPA and enable OVPA to use the HTTP data communication protocol. However, if you want to use the legacy DCE or NCS data communication protocol, the `mwa` script is provided for backward compatibility.

The following table lists the different services that are started for the different protocols.

Table 2 OVPA services started for different protocols

Services started for HTTP protocol	Services started for DCE or NCS protocol
`scopeux`	`scopeux`
`coda`	`coda`
`perfalarm`	`perfalarm`
`midaemon`	`midaemon`
`ttd`	`ttd`
`ovc`	`ovc`
`ovbbccb`	`ovbbccb`
	`llbd (NCS mode)`

Services started for HTTP protocol	Services started for DCE or NCS protocol
	perflbd
	rep_server
	alarmgen (if perfalarm is not present)

The perflbd, rep_server, and alarmgen processes are used for DCE communication

Before you start OV Performance Agent, check to see if any processes are running by typing:

```
/opt/perf/bin/perfstat
```

Using the ovpa script

To start OV Performance Agent and its processes using ovpa:

1. Log in as user **root**.
2. Type: **/opt/perf/bin/ovpa start**

   The ovpa start script starts OV Performance Agent and all its processes, including the scopeux (data collector), midaemon (measurement interface daemon), ttd (transaction tracking daemon), coda, ovc, ovbbccb and the alarm generator. As the script executes, the status of the processes that are started is displayed on the screen.

You can stop OV Performance Agent processes while they are running and restart them using the ovpa script and appropriate options.

- `ovpa stop` stops all OV Performance Agent processes except `ttd` (the transaction tracking daemon), `ovc` and `ovbbccb`. These processes must always be left running. If OpenView Operations agent is running on the system, `ovpa stop` does not stop the `coda` daemon.

If you must stop `ttd`, any ARM-instrumented applications that are running must also be stopped before you restart `ttd` and OV Performance Agent processes.

Individual components can be reinitialized as well with the `ovpa restart` option. Changes to configuration files will *not* take effect on your system unless the corresponding process is restarted.

- `ovpa restart server` causes `coda` to stop and then start, temporarily disabling alarming and access for clients such as OV Performance Manager, and rereads the `datasources` file. It also stops and then restarts the `perfalarm` process and rereads the `alarmdef` file.
- `ovpa restart` causes `scopeux` and the server processes to temporarily stop and then start. It reads the `parm` file as well as forces the transaction daemon `ttd` to reread its configuration file `ttd.conf`.
- `ovpa restart alarm` causes the `perfalarm` process to temporarily stop and then start and reread the `alarmdef` file, so that if you have made changes to the file, the new alarm definitions will take effect without restarting all OVPA processes. This action does *not* disrupt any other processes.

Using the mwa script

To start OV Performance Agent and its processes:

1. Log in as user **`root`**.
2. Type:

   `/opt/perf/bin/mwa [-ncs | -dce] start`

If you are running OV Performance Agent supporting a DCE communication protocol and you want to change to NCS, you have to edit the `/etc/default/ovpa` file and set the `MWA_PROTOCOL` variable to `ncs` and the `MWA_LLBD_COMMAND` to `/opt/perf/bin/llbd`. Otherwise, you will get OV Performance Agent running the `dced` daemon, that emulates the NCS local location broker. You may also need to kill the `dced` daemon before running `./ovpa start`. To confirm that `llbd` is really running, run:

`ps -ef | grep llbd`.

If OV Performance Agent is started in the normal boot sequence, after the startup of all HP OpenView products (OV Performance Agent, OVO, and so on) to ensure that proper communication services have been enabled.

The `mwa` script starts OV Performance Agent and all its processes, including `scopeux` (data collector), `midaemon` (measurement interface daemon), `ttd` (transaction tracking daemon), `coda`, `ovc`, `ovbbccb`, `perflbd`, `rep_server`, and the alarm generator. As the script executes, the status of the processes that are started is displayed on the screen.

You can stop OV Performance Agent processes while they are running and restart them using the `mwa` script and its appropriate options.

- `mwa stop` stops all OV Performance Agent processes except `ttd` (transaction tracking daemon), `ovc`, and `ovbbccb`. `ttd` should always be left running. If OpenView Operations agent is running on the system, `mwa stop` does not stop the `coda` daemon.
- OV Performance Agent can be reinitialized using the `mwa restart` option. Changes to configuration files will *not* take effect on your system unless the corresponding processes are restarted.
- `mwa restart server` causes `coda` and the repository servers to stop and then start, temporarily disabling alarming and access for clients such as OV Performance Manager, and rereads the `perflbd.rc` file. It also stops and then starts the alarm generator process and rereads the `alarmdef` file. The HTTP based alarm generator, `perfalarm`, is enabled by default.

- `mwa restart` causes the server processes and all the daemons including `coda`, `scopeux` and the transaction daemon `ttd` to temporarily stop and then start. It rereads the `parm` file and the `ttd.conf` transaction configuration file.

If you stop `ttd`, any ARM-instrumented applications that are running must also be stopped before you restart `ttd` and OV Performance Agent processes.

- `mwa restart alarm` causes the alarm generator process to temporarily stop and then start and reread the `alarmdef` file. This action does not disrupt any other process.

Changing Protocols

During first time installation of OV Performance Agent, the protocol selected is written to the `/etc/default/ovpa` file as an addition to the environment variable `MWA_PROTOCOL` (for example, `MWA_PROTOCOL=http`). See The /etc/default/ovpa File on page 27, for more information on the `ovpa` configuration file.

To switch to DCE or NCS mode:

You cannot use the `ovpa` script to start or stop DCE or NCS data communication components. You must set the `MWA_PROTOCOL` parameter in the `/etc/default/ovpa` file to `dce` or `ncs` respectively, and start OVPA using the `mwa` script.

To switch to HTTP mode:

If you want to switch to the HTTP protocol, you must set the `MWA_PROTOCOL` parameter in the `/etc/default/ovpa` file to `http` and restart OVPA. To start OVPA using the HTTP data communication mode, you can use either the `ovpa` or `mwa` script.

If `MWA_PROTOCOL` is set to `http`, both the `ovpa` and `mwa` scripts start the same components.

The `mwa` script starts the HTTP, DCE, or NCS data communication components depending on the value set for `MWA_PROTOCOL`.

The HTTP communication protocol is always enabled, irrespective of the protocol you have selected during installation. The daemons used for HTTP data communication are always installed and active on your system.

Starting and Stopping Automatically

The process of starting OV Performance Agent automatically whenever the system reboots and to stop when the system shuts down is controlled by the file `/etc/init.d/ovpa`.

To disable OV Performance Agent to start or stop automatically, edit the file `/etc/default/ovpa` and change the line `MWA_START=1` to `MWA_START=0`. For more information about the `/etc/default/ovpa` file, see The /etc/default/ovpa File on page 27..

The files `/etc/init.d/ovpa` as well as `/etc/default/ovpa` are removed when OV Performance Agent is removed from a system and are *not* overwritten when OV Performance Agent is patched or updated.

The `dced` and `llbd` daemons cannot run at the same time. Make sure the `/etc/inittab` file starts only the daemon you need.

The /etc/default/ovpa File

The `/etc/default/ovpa` file is available with OV Performance Agent. The file contains various environment variables that control the behavior of OV Performance Agent when starting it. The file is a source file for the following scripts:

- `/opt/perf/bin/ovpa` OV Performance Agent control script
- `/etc/init.d/ovpa` OV Performance Agent auto-start script

The file is removed only when OV Performance Agent is removed from a system and is *not* overwritten when OV Performance Agent is updated. When OV Performance Agent is updated, a copy of the default `/etc/default/ovpa` file is left in the `/opt/perf/newconfig` directory under the name `ovpa.default` so that your customized copy does *not* get affected.

The environment and shell variables that can be modified to change the default behavior of OV Performance Agent are listed below.

- `MWA_START` controls the auto-start of OV Performance Agent whenever your system reboots. The variable can have one of the following values:

0	do *not* start OV Performance Agent at the system boot
1	start OV Performance Agent at the system boot

- `MWA_PROTOCOL` determines whether OV Performance Agent servers register their interfaces as NCS or DCE in addition to HTTP. By default, in a first time installation, the variable is set to `http`, and can be one of the following values:

`http`	run OV Performance Agent as an HTTP service
`ncs`	run OV Performance Agent as an NCS service
`dce`	run OV Performance Agent as a DCE service. Use only lower case letters to designate `ncs` or `dce`.

- `The MWA_START_COMMAND` contains a variable that is used to start OV Performance Agent whenever your system reboots. Normally, the variable is set to `/opt/perf/bin/mwa start`.
- The `MWA_LLBD_COMMAND` contains the command string to start the NCS local location broker daemon. The `dced` daemon is capable of emulating the NCS local location broker. However, if you still want to run the genuine NCS `llbd` you can set the command to:

  `MWA_LLBD_COMMAND="/opt/perf/bin/llbd"`

The variable only applies when OV Performance Agent operates in NCS mode if `llbd` or `dced` are *not* already running.

- `MWA_RPC_INETADDR` defines the network interface that has to be used, on multi-homed systems running in the NCS mode, for communication with client products, such as OV Performance Manager. NCS does *not* support registration to multiple interfaces, therefore this environment variable must explicitly be set if the default network interface cannot be reached by the client products. The variable's value, *which must be exported*, is the IP address in dotted format. For example:

  `MWA_RPC_INETADDR=127.0.0.1`

```
export MWA_RPC_INETADDR
```

The RPC_RESTRICTED_PORTS environment variable has been moved to the /etc/default/hplwdce file but will remain in the /etc/default/ovpa file as commented text only. The variable will have no effect on the behavior of OV Performance Agent unless set in the /etc/default/hplwdce file.

The /etc/default/hplwdce File

The /etc/default/hplwdce file contains various DCE environment variables that control the behavior of OV Performance Agent on startup.

The hplwdce file sources the /etc/default/ovpa file.

The following variables are DCE environment variables that affect only the behavior of DCE runtime and DCE based servers. Although OV Performance Agent provides basic DCE runtime services as a part of the package, it will automatically detect and deploy any existing Transarc or DASCOM DCE runtime installation. In such cases, some environment variables must also be set in the appropriate startup script of the existing DCE product in order to work correctly. Such environment variables are marked with "Yes" in the DCE runtime line.

If a DCE installation has been detected and is not running, OV Performance Agent will not start and an error message is generated.

RPC_RESTRICTED_PORTS restricts the range from which the DCE runtime selects the communication ports to the listed range. This behavior is useful when a client and server must communicate through a port filtering firewall. Note that the range must not be too small or else the runtime will run out of resources. The RPC_RESTRICTED_PORTS environment variable affects the entire DCE runtime and thus all applications that use DCE. Note that OV Performance Agent services require one communication port for each registered data source plus additional five ports. For example:

```
RPC_RESTRICTED_PORTS=ncadg_ip_udp[xxxx-yyyy]
export RPC_RESTRICTED_PORTS
```

For more information about Firewall support, see Communicating Across Firewall on page 32.

- The RPC_UNSUPPORTED_NETADDRS environment variable is used to prevent binding DCE services to the interfaces listed in the variable string. If you want to exclude network addresses from the DCE binding list, replace the 127.0.0.1 from the example below, with a list of real addresses separated by a colon (:).

  ```
  RPC_UNSUPPORTED_NETADDRS=127.0.0.1
  export RPC_UNSUPPORTED_NETADDRS
  ```

- The RPC_SUPPORTED_NETADDRS environment variable is used to enable the binding of DCE services to the interfaces listed in the variable string. On multi-homed systems it is sometimes desired to enable the use of only certain networks for DCE based services. If you want to include network addresses to the DCE binding list replace the 127.0.0.1 IP address from the example below, with a list of real addresses.

  ```
  RPC_SUPPORTED_NETADDRS=127.0.0.1
  export RPC_SUPPORTED_NETADDRS
  ```

- The RPC_NOALIAS_NETIFS environment variable may have values 1 or 0 (default). If the variable is set to 1 only the primary IP address for each local network interface is extracted as the usable set of network addresses for this DCE client. All IP aliases for all network interfaces are ignored. This variable is automatically set to 1 in environments where the number of IP addresses is greater than 32.

Status Checking

Several status files are created in the /var/opt/perf/ and /var/opt/OV/ directories when OV Performance Agent is started. You can check the status of all or some OV Performance Agent processes using the perfstat command.

The following status files contain diagnostic information you can use to troubleshoot problems that may arise with the OV Performance Agent processes.

```
/var/opt/perf/status.alarmgen
/var/opt/perf/status.perflbd
/var/opt/perf/status.rep_server
/var/opt/perf/status.scope
/var/opt/perf/status.perfalarm
/var/opt/perf/status.ttd
/var/opt/perf/status.mi
/var/opt/OV/log/coda.txt
```

Every time an OV Performance Agent process writes a message to its status file, it checks to see if the file is larger than one MB. If it is, the file is renamed to `status.filename.old` and a new status file is created.

Examples Directory

The `/opt/perf/examples` directory contains examples of configuration files, syntax files, and sample program files that can be used to customize your HP Performance Tools. For example, the `/opt/perf/example/ovpaconfig/` subdirectory contains sample alarm definitions and examples of `parm` file application-specific parameters. For more information, see the `/opt/perf/examples/README` file.

Communicating Across Firewall

A firewall can be defined as a method for filtering the flow of data between one network and another. OV Performance Agent (OVPA) now supports HTTP 1.1 based communications interface for data access between clients such as OV Performance Manager and OV Reporter and server applications, in addition to the previously supported communication mechanism through a packet-filtering network firewall.

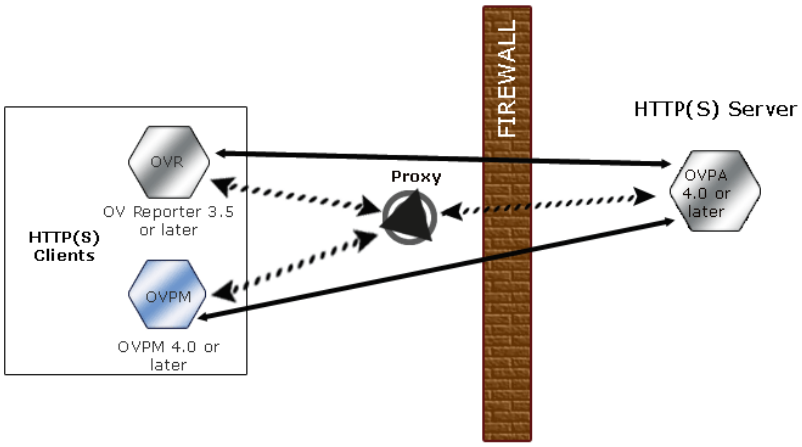
OV Performance Agent supports certificate-based secure (HTTPS) data communication only in the HP OpenView Operations 8.x environment. For more information, see Using Certificates on page 47.

The HTTP based interface is flexible, because it can use proxies, requires fewer ports and is firewall friendly. The DCE interface is still available for use.

The following sections explain how to configure HTTP and DCE communication across a firewall:

- Communicating in the HTTP Environment
- Communicating in the DCE Environment

Figure 1 Communicating with OVPA in a Firewall Environment



The name OV Performance Manager 3.x is used throughout this document to refer to the product that was formerly known as PerfView.

Communicating in the HTTP Environment

For firewall configuration it is important to know which system initiates the communication (client) and which receives communication requests (server), so that the firewall rules can be set up accordingly. In a typical remote communication, a client, using the source port, connects to a server that is listening on the destination port on a remote system. Understand your firewall environment including the client and server data flow. To configure communications with OVPA in a firewall environment, perform the following tasks:

1 Configure OVPA Ports.

2 Configure HTTP Clients in a Firewall Environment.

3 Verify Firewall Configuration.

Figure 1 on page 33 shows how OVPA communicates with Reporter (version 3.5 or later) and OV Performance Manager (OVPM version 4.0 or later) through a firewall. OVPA is an HTTP or HTTPS server. Reporter and OVPM 4.x are HTTP clients. OVPM 5.0 can be an HTTP or HTTPS client. If an HTTP proxy is used, Reporter and OVPM communicate with OVPA through the proxy.

OVPM version 3.x uses the PerfView technology. PerfView does not use the HTTP datacomm components, but it connects to OV Performance Agent 4.5 when the DCE data communication mode is enabled.

Configure OVPA Ports

You can configure OVPA ports in a firewall environment in one of the two ways:

- Configure Two-Port Communication
- Configure Single-Port Communication
- Verify Port Settings

On an OVPA system using BBC5, by default, the BBC communication broker uses port 383 and `coda` uses a dynamically allocated port.

Configure Port Settings for the BBC Communication Broker

You can configure the port settings of the default port used by the BBC communication broker. Use the `ovconfchg` command to change the port settings on the OV Performance Agent system. You can use one of the following options:

— **ovconfchg -ns bbc.cb.ports -set PORTS <*host*>:<*port*>**

(Or)

— **ovconfchg -ns bbc.cb.ports -set PORTS <*domain*>:<*port*>**

Example: **ovconfchg -ns bbc.cb.ports -set PORTS xyz.abc.com:50383**

Restart ovpa using the following command:

```
ovpa restart server
```

Configure Two-Port Communication

By default, `coda` deamon uses a dynamically chosen second port, in addition to port 383 used by the BBC communication broker. You can configure the port settings of `coda` to listen at a well known port of your choice using the `ovconfchg` command. Type the following commands:

`ovconfchg -ns coda.comm -set SERVER_PORT <`*portnumber*`>`

`ovpa restart server`

Using a dynamic port when connecting to OVPA remotely through a firewall can be difficult, because you may not know the firewall ports to open.

Configure Single-Port Communication

On the OV Performance Agent system, the BBC communication broker uses port 383 and `coda` uses a port that is dynamically allocated. You can configure the port settings for `coda` to share the same port used by the communication broker using the `ovconfchg` command. Type the following commands:

`ovconfchg -ns coda.comm -set SERVER_BIND_ADDR localhost`

`ovpa restart server`

Verify Port Settings

To verify the port settings, type the following command:

`perfstat -d`

The output displays the following information:

- port number of the port used by `Coda`
- port number of the port used by `BBC communication broker`
- the port settings configured
- indicates if secure communication is enabled
- indicated if `coda` metric collection is enabled

For example:

```
Datacomm configuration :
------------------------
 Coda Port                          49552 (Dynamic)
```

```
                                        Two port Communication
BBC communication broker port                383
SSL security                                 NONE
Coda Metric Collection(Prospector)           Disabled
```

Configure HTTP Clients in a Firewall Environment

There are two ways to configure HTTP clients in a firewall environment:

- Configuring HTTP Clients (Reporter/OVPM) with HTTP Proxy
- Configuring HTTP Clients (Reporter/OVPM) without HTTP Proxy

In both cases, to access data from OVPA nodes, only one port needs to be opened on the HTTP server (OVPA) side.

Configuring HTTP Clients (Reporter/OVPM) with HTTP Proxy

It is recommended that you use HTTP proxies when communicating through a firewall. This simplifies the configuration by using proxies that are often already in use in your environment. The firewall must be open for exactly one port if proxies are to be used in both directions. To access data collected by OVPA, ports for the HTTP server (OVPA) and the HTTP client (Reporter and OVPM) must be opened.

It is recommended that you do not change the default 383 port.

When an HTTP proxy is used, Reporter and/or OVPM for Windows and UNIX need to be configured to specify the proxy to be used to contact OVPA.

To configure OVPM versions 5.0 and later, and Reporter 3.7:

Type the following command,

`ovconfchg -ns bbc.http -set PROXY proxy:port+(a)-(b)`

The variables *a* and *b* are comma separated lists of hostnames, networks, and IP addresses that apply to the proxy. Multiple proxies may be defined for one PROXY key using the "*;*" or "*,*" delimiter. "–" before the list indicates that those entities do not use this proxy, "+" before the list indicates that those entities do use this proxy. The first matching proxy is used.

To configure Reporter versions 3.6 and earlier, and OVPM 4.x:

Edit the `/var/opt/OV/conf/BBC/default.txt` configuration file.

In the [DEFAULT] section of the default.txt file, locate the lines that relate to the PROXY and set the PROXY parameter as follows.

```
PROXY web-proxy.hp.com:8088-(localhost, *.hp.com) + (*)
```

In this example, the proxy web-proxy will be used with port 8088 for every server (*) except requests for the local machine (localhost) and requests internal to HP (matching *.hp.com, for example **www.hp.com**.

Configuring HTTP Clients (Reporter/OVPM) without HTTP Proxy

If HTTP proxies are not available, additional configuration settings are required on the HTTP clients (Reporter and OVPM system).

If Reporter and OVPM for Windows are installed on the same system and both access OVPA in parallel, you can specify a port range as described in this section. If they are running on different systems, you can specify a single port for each. Depending on the versions of OVPM and Reporter you are using select from the following options:

Configure OVPM 5.0 and later, and Reporter 3.7 as follows:

Type the following command,

ovconfchg -ns bbc.http -set CLIENT_PORT <*port range*>

Where <*port range*> is the range of ports you want to use.

For example:

```
ovconfchg -ns bbc.http -set CLIENT_PORT 14000-14003
```

Configure Reporter versions 3.6 and earlier, and OVPM 4.x as follows:

Edit the /var/opt/OV/conf/BBC/default.txt file as follows.

1. Locate the lines that apply to CLIENT_PORT and uncomment the line ;CLIENT_PORT =.
2. Specify the port range for the CLIENT_PORT parameter. For example:

   CLIENT_PORT = <*port range*>

   Where <*port range*> is the range of ports you want to use. For example:

   CLIENT_PORT = 14000-14003

Verify Firewall Configuration

To verify your configuration, use the command:

ovcodautil -ping -n <*system name*>

This output of this command indicate the status of your communication settings.

Configuring Systems with Multiple IP Addresses

If your environment includes systems with multiple network interfaces and IP addresses and you want to use a dedicated interface for the HTTP-based communication, then you can use the parameters `CLIENT_BIND_ADDR` and `SERVER_BIND_ADDR` to specify the IP address that should be used.

- If you have multiple network interfaces and IP addresses on the OVPA (Server) system, specify the `SERVER_BIND_ADDR` parameter as follows:

  ovconfchg -ns bbc.http -set SERVER_BIND_ADDR <IP Address>

- If you have multiple network interfaces and IP addresses on the OVPM 5.0 (client) system, specify the `CLIENT_BIND_ADDR` parameter as follows:

  ovconfchg -ns bbc.http -set CLIENT_BIND_ADDR <IP Address>

- If you have multiple network interfaces and IP addresses on the Reporter/ OVPM 4.x system, specify the `CLIENT_BIND_ADDR` parameter.

  Edit the `/var/opt/OV/conf/BBC/default.txt` file as follows:

  a. Locate the lines that apply to `CLIENT_BIND_ADDR` and uncomment the line

     ```
     ;CLIENT_BIND_ADDR =
     ```

  b. Specify the IP address for the `CLIENT_BIND_ADDR` parameter.

Communicating in the DCE Environment

In the DCE environment, OV Performance Agent uses dynamically allocated socket port numbers for interprocess communication. To communicate through a packet-filtering network firewall, you must configure the OV Performance Agent servers to use statically defined port numbers.

Configuring OV Performance Manager and OV Performance Agent Communication

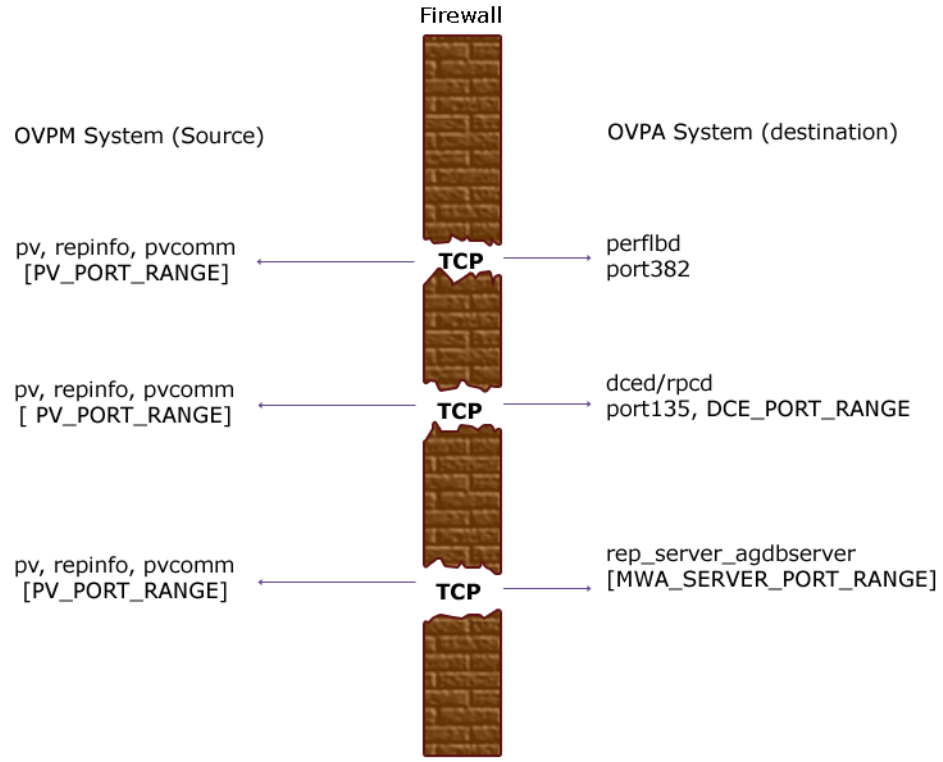
The method of configuring the socket port numbers for OV Performance Agent depends on the version of OV Performance Manager that will be communicating with OV Performance Agent. To configure the socket port numbers, follow the instructions in Configuring OV Performance Manager C.03.00 and Later with OV Performance Agent Communication on page 39.

Configuring OV Performance Manager C.03.00 and Later with OV Performance Agent Communication

The way to configure OV Performance Agent and OV Performance Manager firewall communication depends on which of these two programs is the source of the communication.

Configuring OVPM C.03.00 and Later (Source) with OVPA Communication

When OV Performance Manager is the source, it communicates with OV Performance Agent using the TCP protocol with the TCP socket port numbers shown in the following figure.



To configure the `MWA_SERVER_PORT_RANGE` as statically defined TCP socket port numbers, add the following entries to the `/etc/services` file:

`agdbserver` *xxxx*`/tcp`
`rep_server` *yyyy*`/tcp`

where *xxxx* and *yyyy* specify unused port numbers. `agdbserver` and `rep_server` register at the specified port numbers. If there are multiple data sources configured in the `perflbd.rc` file, the first `rep_server` uses the *yyyy* port number specified above. All other `rep_servers` add one to the last used port number.

For example, if you include the following lines in the `/etc/services` file:

```
agdbserver    20001/tcp
rep_server    20002/tcp
```

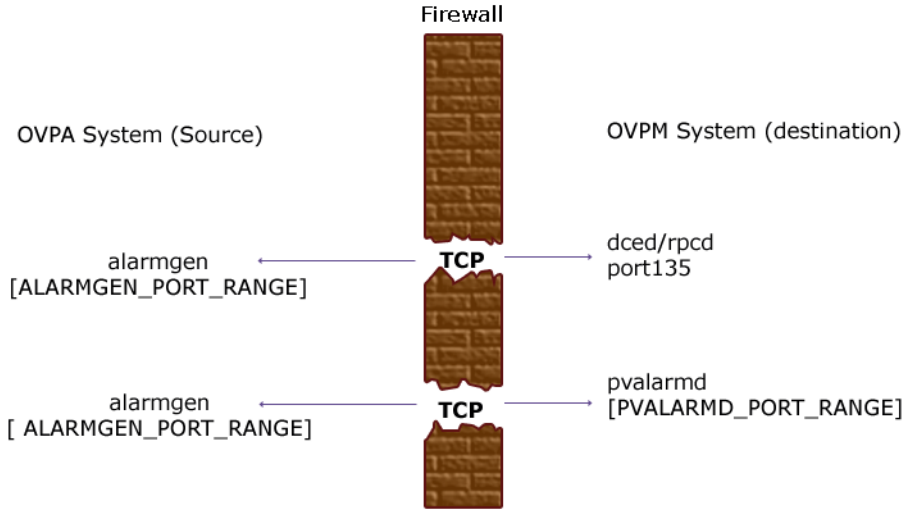
and there are three data sources configured in the `perflbd.rc` file, OV Performance Agent will use the following TCP port numbers:

```
agdbserver    20001
rep_server    20002
rep_server    20003
rep_server    20004
```

Based on the `/etc/services` example above, the `MWA_RANGE` will be 20001-20004.

Configuring OVPA (Source) with OVPM C.03.00 and Later Communication

When OV Performance Agent is the source, it communicates with OV Performance Agent using the TCP protocol with the TCP socket port numbers shown in the following figure.



- If you previously had OV Performance Agent communicating to OV Performance Manager through a firewall, port 135/UDP was open in the firewall. The firewall configuration must now be changed to open port 135/TCP. This is because the protocol used to connect to port 135 on the OV Performance Manager system was changed from UDP to TCP, regardless of the version of OV Performance Manager.

  To configure the `ALARMGEN_PORT_RANGE` for the `alarmgen` process, edit the file `/var/opt/perf/vppa.env` and set the `RPC_RESTRICTED_PORTS` to the following:

  `RPC_RESTRICTED_PORTS=ncacn_ip_tcp [`*`xxxx-yyyy`*`]`

  where *xxxx-yyyy* is a range of unused port numbers. The formula for calculating the port range is:

  ```
  2 *  (# of OVPM systems receiving alarms from the OV Performance
       Agent system)
  ```

  For example, if the OV Performance Agent system was sending alarms to two OV Performance Manager systems, set `RPC_RESTRICTED_PORTS` to the following range in the `vppa.env` file:

  `RPC_RESTRICTED_PORTS=ncacn_ip_tcp[30001-30004]`

- This environment variable affects only the ports that are used for communication outside the localhost. Ports that are used internal to the local host, such as local communication between `alarmgen` and `rep_server`, are not affected by this variable.

  Restart the OV Performance Agent servers using `/opt/perf/bin/mwa restart server` to make the port restriction take effect.

  Refer to the *HP OpenView Performance Manager Installation Guide* to determine the PV_PORT_RANGE.

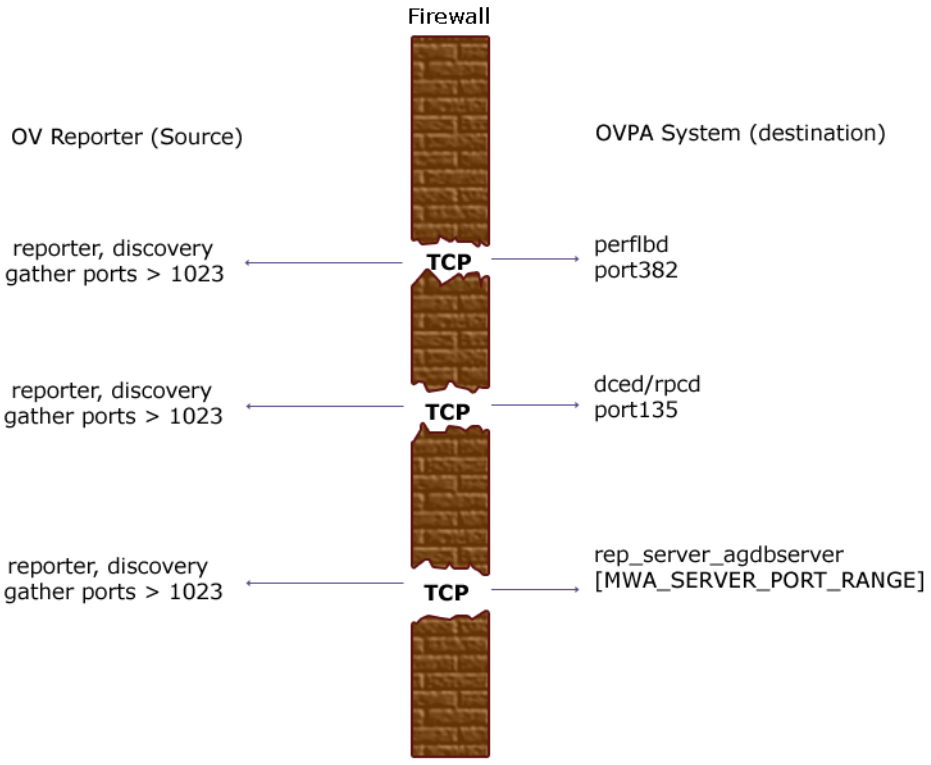
- Sending alarms from OV Performance Agent to OV Performance Manager through firewalls that use Network Address Translation (NAT) IP addresses is not supported.

Configuring OV Reporter and OV Performance Agent Communication

The configuration of the socket port numbers for OV Performance Agent depends on the version of OV Reporter that will be communicating with OV Performance Agent. To configure the socket port numbers, follow the instructions in Configuring OV Reporter A.03.50 and Later with OV Performance Agent Communication.

Configuring OV Reporter A.03.50 and Later with OV Performance Agent Communication

OV Reporter communicates with OV Performance Agent using the TCP protocol with the TCP socket port numbers shown in the following figure.



For instructions on configuring the `MWA_SERVER_PORT_RANGE` as statically defined TCP socket port numbers, see Configuring OV Performance Manager C.03.00 and Later with OV Performance Agent Communication on page 39.

Restricting RPC Addresses

The `RPC_SUPPORTED_NETADDRS` environment variable is used to force the DCE/RPC service to bind to specific LAN cards in a multi-homed environment. If this environment variable is set, only addresses in the list are advertised in the endpoint map; addresses not found on the list are excluded from the server's list of available addresses.

To restrict the servers to using only a specified set of IP addresses, set the RPC_SUPPORTED_NETADDRS environment variable before starting the servers.

The syntax is:

RPC_SUPPORTED_NETADDRS=protocol:ip_address[,protocol:ip_address]

To set the environment variable for the OV Performance Agent servers, add the lines:

```
RPC_SUPPORTED_NETADDRS=ncadg_udp:192.1.1.1
export RPC_SUPPORTED_NETADDRS
```

to the /var/opt/perf/vppa.env file and then restart the servers using mwa restart server.

If you still have problems connecting, the problem may be with the endpoint map (dced/rpcd). Try setting the environment variable *before* starting dced/rpcd. Then restart the system to make the IP address restriction take effect.

Configuring Port Ranges for OV Performance Agent

Set the RPC_RESTRICTED_PORTS environment variable as follows:

- Stop the OV Performance Agent server.

  /opt/perf/bin/mwa stop server

- Stop the dced daemon.

  /etc/init.d/hplwdce stop

- Determine a suitable port range. To do so use the following formula:

  ```
  MWA_PORT_RANGE=(# rep_server)*7 + (# pvalarmd)*2+10
  ```

- Edit the /etc/default/hplwdce.
  Uncomment the following lines:

  ```
  RPC_RESTRICTED_PORTS=ncadg_ip_udp[xxxx-yyyy] \
  :ncan_ip_tcp[xxxx-yyyy]
  export RPC_RESTRICTED_PORTS
  ```

where [*xxxx-yyyy*] represents the range of addresses you have chosen. The default recommended range is [10500-10540].

- Start the dced daemon.

  /etc/init.d/hplwdce start

- Start OV Performance Agent, run:

  `/opt/perf/bin/mwa start server`

Refer to the *HP OpenView Performance Manager (PerfView) Installation Guide* to determine the `PV_PORT_RANGE`.

Configuring Secure Communication

OV Performance Agent supports certificate-based secure communication and client authentication based communication.

Using Certificates

OV Performance Agent supports certificate-based secure data communication only in the HP OpenView Operations 8.x environment.

To configure secure communication on your OVO setup, refer to the *HP OpenView Operations for UNIX Firewall Concepts and Configuration Guide*. For more information on OVO 8.x HTTPS agent, refer to the *HP OpenView Operations HTTPS Agent Concepts and Configuration Guide*.

If you have already configured HTTPS communication in the OVO 8.x environment, make the following changes to configure secure communication between OV Performance Agent and OVPM 5.0.

OV Reporter and OVPM 4.x do not support certificate-based secure communication.

On the OV Performance Agent system, set `SSL_SECURITY` to `REMOTE` for `coda`. Type the following commands:

```
ovconfchg -ns coda -set SSL_SECURITY REMOTE

ovcodautil -config
```

Using Client Authentication

OV Performance Agent enables optional authentication of client connections from products such as, OV Performance Manager or Reporter (Service Reporter). The authentication capability allows you to specify, for a given OV Performance Agent instance, which hosts are allowed to make client connections to that instance.

The Client Authentication feature enables/disables connections from any version of the OV Performance Manager and Reporter clients. Your client software does *not* need to be updated for you to take advantage of this feature.

For authorized clients the authentication process is transparent, their client connection proceeds as it has with previous versions of OV Performance Agent. Unauthorized clients receive a message indicating denial of service, for example:

```
Could not connect to OV Performance Agent data source on host
<hostname>.
```

The unauthorized connection attempt is logged in the `status.rep_server` file as the following message:

```
UNAUTHORIZED CONNECTION ATTEMPT:<IP address of connecting host in
dotted quad format> (MWA201-16)
```

Enabling Authentication with the authip File

Authentication is enabled by the presence of a file called `authip`. On systems where HTTP communication is enabled, the `authip` file exists in the `/var/opt/OV/conf/perf/` directory. On systems where DCE communication is enabled, the `authip` file exists in the `/var/opt/perf/` directory. The `authip` file lists hosts from which client connections are to be permitted.

- If the `authip` file exists in the default directory, then its contents determine which hosts are permitted client connections. Clients running on the same host as the OV Performance Agent instance are automatically authenticated, which means the clients do *not* need an entry. A zero-length `authip` file dictates that only clients running on the OV Performance Agent host can connect.
- If the `authip` file does *not* exist in the default directory, then no authentication is performed and any client will be allowed to connect, as was the case with prior OV Performance Agent versions.

The `authip` file is checked each time a client attempts to register for service with OV Performance Agent. OV Performance Agent does *not* need to be restarted for changes to the `authip` file to become effective.

Note, however, that an existing authorized client session can continue its current connection despite a subsequent change in the server's `authip` file, which would otherwise disqualify it, until the client takes an action that requires re-registration with OV Performance Agent. Thus, an authorized OV Performance Manager connection continues to be permitted, regardless of changes in the OV Performance Agent `authip` file, until the data source to the OV Performance Agent host has been closed. If there is then an attempt to reopen the data source, the `authip` file is reread and the connection is denied.

In the case of OV Performance Manager registration for alarms, a previously authorized client will continue to receive alarms until the data source has been removed (not just closed) by the client. If you want to force removal of a client from the server's alarm generator database from the OV Performance Agent side, use the command:

```
agsysdb -delpv <host>
```

The OV Performance Agent client authentication capability requires that your network be able to resolve the client entries in the `authip` file. Depending upon the nature of the entries, this may require name services such as those provided by DNS, NIS, or `/etc/hosts` files.

A good test is to ensure that you can successfully "ping" each `authip` entry from the OV Performance Agent host. Client authentication works through a firewall with the same proviso that the client entries in the `authip` file be pingable from the OV Performance Agent host.

Formatting the authip File

The `authip` file must conform to the following format:

- One client host may be listed per line.
- Client entries can be in any one of the following formats:
  - Fully qualified domain name
  - Alias
  - IP address (must be in IPv4 dotted quad format)
- Client entries can have no embedded spaces.
- A line containing a # in the first column is taken as a comment, and is ignored.
- Blank or zero-length lines are ignored.
- The IP address may *not* have a leading zero. For example, the IP address 23.10.10.10 cannot be represented as `023.10.10.10`.

  Thus, given an `/etc/hosts` entry as follows:

  ```
  123.456.789.1 testbox testbox.group1.thecompany.com
  ```

  any one of the following entries in the `authip` file would enable clients from the `testbox` host to connect:

```
#============ Examples of authip file entries ==============
#
# Use of an IP address
123.456.789.1

#
# Use of an alias
testbox

#
# Use of a fully qualified domain name
testbox.group1.thecompany.com

#============ End of examples of authip file entries ========
```

Configuring Data Sources

OV Performance Agent uses the coda daemon or a set of repository servers that provide previously collected data to the alarm generator and the OV Performance Manager analysis product. The coda daemon uses the HTTP data communication mechanism, and the repository servers use the DCE mechanism. If both HTTP and DCE data communication mechanisms are enabled, OVPA uses both the coda daemon and the set of repository servers. Each data source consists of a single log file set.

The data source list that coda accesses is maintained in the datasources configuration file that resides in the /var/opt/OV/conf/perf/ directory. The data source list that the repository servers access is maintained in the perflbd.rc file that resides in the /var/opt/perf/ directory. The perflbd.rc file is maintained as a symbolic link to the datasources file.

There is a repository server for each specific data source such as scopeux log files or DSI log files. When you first start up OV Performance Agent after installation, a default data source named SCOPE is already configured and provides a scopeux log file set.

If you want to add other data sources, you can configure them in the datasources file. If you no longer want to view the OVPA or DSI log file data from OV Performance Manager, or process alarms for the log file, you can modify the datasources file to remove the data source and the path to the log file set. When you restart the coda daemon or the repository server, it reads the datasources file and makes the data available over datacomm linkages to analysis tools for each data source it finds. Restart coda or the repository server as described in Datasources Configuration File Format on page 52.

You can also remove the log file set if you no longer need the data. If you remove the log file set but do not remove the data source from datasources, coda or the repository server will skip the data source.

You might also choose to stop logging DSI data to a log file set but keep the coda daemon or the repository server open so you can view the historical data in OV Performance Manager. In this case, stop the dsilog process but do not delete the data source from the datasources file.

Datasources Configuration File Format

Each entry you place into the `datasources` configuration file represents a data source consisting of a single log file set. The entry specifies the data source name and location. Fields are case-insensitive except for the log file path name. The syntax is:

`datasource=datasource_name logfile=logfile_set`

- **`datasource`** is a keyword. **`datasource_name`** is the name used to identify the data source. For example, the data source name used in alarm definitions or by analysis software. Data source names must be unique. They are translated into upper case. The maximum length for a data source name is 64 characters.
- **`logfile`** is a keyword. **`logfile_set`** is the fully-qualified name identifying the DSI log file (created by the `dsilog` process, ending in `.log`), and is case-sensitive.

Following are two examples of the `datasources` file's data source entries:

```
datasource=SCOPE logfile=/var/opt/perf/datafiles/logglob
datasource=ASTEX logfile=/tmp/dsidemo/log/astex/ASTEX_SDL
```

After updating `datasources`, run the following command to make the new data sources available through `coda`:

`/opt/perf/bin/ovpa restart server`

If you are also running repository servers, run the following command to make the new datasources available through repository servers (`rep_server`):

`/opt/perf/bin/mwa restart server`

Note that stopping repository server processes results in any connection to OV Performance Manager being lost. For example, if you are drawing a graph on a data source and try to draw another graph, you will need to reselect the data source in OV Performance Manager and re-establish the connection when the repository server is started again.

Examine the contents of the `/var/opt/OV/log/coda.txt` file to check if the `coda` daemon was activated or for error messages.

For specific examples of configuring DSI data sources, see "Configuring Data Sources" in Chapter 4 of the *HP OpenView Performance Agent for UNIX Data Source Integration Guide*.

Parm File

The `parm` file is a text file that specifies configuration of the `scopeux` data collector including log file maximum sizes, interesting process threshold definitions, and application definitions. Comments in the file provide an overview of the various settings.

The `parm` file is provided with OV Performance Agent in the `/opt/perf/newconfig/` directory and is copied into the `/var/opt/perf/` directory during installation, if there is not an existing `/var/opt/perf/parm` file. For a complete description of the `parm` file and its parameters, see the "Parm File" section in Chapter 2 of the *HP OpenView Performance Agent for UNIX User's Manual*.

Defining Alarms

If you plan to use alarms to monitor performance, you need to specify the conditions that generate alarms in a set of alarm definitions in the OV Performance Agent `alarmdef` file. When OV Performance Agent is first installed, the `alarmdef` file contains a set of default alarm definitions. You can use these default definitions or customize them to suit your needs.

For instructions on defining alarms, see Chapter "Performance Alarms," in your *HP OpenView Performance Agent for UNIX User's Manual*. This chapter also describes the alarm definition syntax, how alarms work, and how alarms can be used to monitor performance.

Viewing and Printing Documents

OV Performance Agent software includes the standard OV Performance Agent documentation set in viewable and printable file formats. You can view the Adobe Acrobat format (`*.pdf`) documents online and print as needed. ASCII text (`*.txt`) documents are printable. However, you can view a text file on your screen using any UNIX text editor such as `vi`.

The documents are listed in the following table along with their file names and online locations.

Table 3 Printables

Document	File Name	Location
HP OpenView Performance Agent for Sun Solaris Systems Installation & Configuration Guide	`ovpainst.pdf`	`/opt/perf/paperdocs/ovpa/C/`
HP OpenView Performance Agent for UNIX User's Manual	`ovpausers.pdf`	`/opt/perf/paperdocs/ovpa/C/`
HP OpenView Performance Agent for UNIX Data Source Integration Guide	`ovpadsi.pdf`	`/opt/perf/paperdocs/ovpa/C/`
HP OpenView Performance Agent for UNIX & GlancePlus Tracking Your Transactions	`tyt.pdf`	`/opt/perf/paperdocs/ovpa/C/`
Application Response Measurement (ARM) API Guide	`arm2api.pdf`	`/opt/perf/paperdocs/arm/C/`
HP OpenView Performance Agent Sun Solaris Metric Definitions	`metsun.txt`	`/opt/perf/paperdocs/ovpa/C/`
OVPA metrics list by Data Class for all operating systems	`mettable.txt`	`/opt/perf/paperdocs/ovpa/C/`

Viewing Documents on the Web

The listed documents can also be viewed on the HP OpenView Manuals web site at:

http://ovweb.external.hp.com/lpe/doc_serv

Select **Performance Agent** from the product list box, select the release version, select the OS, and select the manual title. Click **[Open]** to view the document online, or click **[Download]** to place the file on your computer.

Adobe Acrobat Files

The Adobe Acrobat files were created with Acrobat 7.0 and are viewed with the Adobe Acrobat Reader versions 4.0 and later. If the Acrobat Reader is *not* in your Web browser, you can download it from Adobe's web site:

http://www.adobe.com

While viewing a document in the Acrobat Reader, you can print a single page, a group of pages, or the entire document.

You can read a `.PDF` using the `acroread` command, if you have installed the Adobe Acrobat Reader on your system. Enter the following command where <*path*> is the location of the `acroread` command.

<*path*>**/acroread** <*filename*>`.pdf`

ASCII Text Files

To print a `.txt` file, type:

lp -dprintername *filename*

For example,

lp -dros1234 metsun.txt

A Configuring Coda

Introduction

This appendix provides a list of options to configure `coda` and the BBC communication broker `ovbbccb`, using the `ovconfchg` tool to change the configuration settings for OV Performance Agent.

DISABLE_PROSPECTOR

Use this option to specify the data collection preferences through coda, when both the OVO agent and OVPA are installed. The default value is `false`. The format is as follows:

`ovconfchg -namespace coda -set DISABLE_PROSPECTOR` *<value>*

- `true`: `coda` will not collect data for the Coda datasource
- `false`: `coda` will collect the data for the Coda datasource

RESPONSE_SIZE_LIMIT

Use this command to specify the maximum amount of memory allocated by the `coda` daemon for a query response. The default value is `104857600 (100 megabytes)`. The format is as follows:

`ovconfchg -namespace coda -set RESPONSE_SIZE_LIMIT` *<value>*

If the specified limit is exceeded the following error message appears:

```
coda_out_of_resource
```

SSL_SECURITY

Use this option to enable secure communication through coda. The default value is NONE. The format is as follows:

ovconfchg -namespace coda -set SSL_SECURITY *<value>*

- NONE: coda does not require SSL connections for either the local or remote clients
- REMOTE: coda requires all remote connections to use SSL
- ALL: coda requires all connections (both local and remote) to use SSL

SERVER_BIND_ADDR

Use this option to specify the bind address for the server port. The default value is localhost. When the value is set to localhost, all clients connect to ovbbccb and requests are forwarded by ovbbccb to coda. The format is as follows:

```
ovconfchg -namespace coda.comm -set SERVER_BIND_ADDR
localhost
```

LOG_SERVER_ACCESS

You can enable or disable the access to server using this option. If this option is set to true, the communcation broker BBC records every access to the server, providing information about the senders IP address, requested HTTP address, requested HTTP method, and response status. This value typically will not be changed.

```
ovconfchg -namespace coda.comm -set LOG_SERVER_ACCESS false
```

PROXY

Use this option to configure the proxy. The format is as follows:

```
ovconfchg -ns bbc.http -set PROXY proxy:port+(a)-(b)
```

Where, the variables *a* and *b* are comma separated lists of hostnames, networks, and IP addresses that apply to the proxy.

Multiple proxies may be defined for one PROXY key using the "**;**" or "**,**" delimiter. "**-**" before the list indicates that those entities do not use this proxy, "**+**" before the list indicates that those entities use this proxy. The first matching proxy is used.

For example:

```
ovconfchg -ns bbc.http -set PROXY srv1.abc.com:8088+*
```

```
ovconfchg -namespace coda.comm -set SERVER_PORT 0
```

SERVER_PORT

You can configure the port settings of the default port used by the the BBC communication broker. Use the `ovconfchg` tool to change the port settings on the OV Performance Agent system. Type the commands:

`ovconfchg -ns bbc.cb.ports -set SERVER_PORT <`*`port number`***`>`**

`ovconfchg -namespace coda.comm -set SERVER_PORT <`*`port number`***`>`**

Glossary

A

alarm

An indication of a period of time in which performance meets or exceeds user-specified alarm criteria. Alarm information can be sent to an analysis system (such as OV Performance Manager) and to OV Operations. Alarms can be identified in historical data log files using the `utility` program.

alarm generator

Handles the communication of alarm information. It consists of `perfalarm` and the `agdb` database. The `agdb` database contains a list of OV Performance Manager analysis nodes (if any) to which alarms are communicated, and various on/off flags that you set to define when and where the alarm information is sent.

alarmdef file

The file containing the alarm definitions in which alarm conditions are specified.

application

A user-defined group of related processes or program files. Applications are defined so that performance software can collect performance metrics for and report on the combined activities of the processes and programs.

application log file

See **logappl**.

C

coda daemon

A daemon that provides collected data to the alarm generator and analysis product data sources including `scopeux` log files or DSI log files. `coda` reads the data from the data sources listed in the `datasources` configuration file.

D

data source

Consists of one or more classes of data in a single `scopeux` or DSI log file set. For example, the OV Performance Agent SCOPE data source is a `scopeux` log file set consisting of global data. See also **datasources file**.

datasources file

A configuration file residing in the `/var/opt/OV/conf/perf/` directory. Each entry in the file represents a `scopeux` or DSI data source consisting of a single log file set. See also **perflbd.rc**, **coda** and **data source**.

data source integration (DSI)

The technology that enables OV Performance Agent to receive, log, and detect alarms on data from external sources such as applications, databases, networks, and other operating systems.

default.txt

A communications configuration file used to customize communication parameters for HP OpenView applications.

device

A device is an input and/or output device connected to a system. Common devices include disk drives, tape drives, CD-ROM drives, printers, and user terminals.

device log file

See **logdev**.

DSI

See **data source integration**.

DSI log files

Log files containing self-describing data that are created by OV Performance Agent's DSI programs.

E

extract

An OV Performance Agent program that allows you to extract (copy) data from raw or previously extracted log files and write it to extracted log files. It also lets you export data for use by analysis programs.

extracted log file

A log file created by the `extract` program. It contains user-selected data ranges and types of data. An extracted log file is formatted for optimal access by the workstation analysis tool, OV Performance Manager. This file format is suitable for input to the `extract` and `utility` programs and is the preferred method for archiving performance data.

G

GlancePlus

GlancePlus (or Glance) is an online diagnostic tool that displays current performance data directly to a user terminal or workstation. It is designed to assist you in identifying and troubleshooting system performance problems as they occur.

global

A qualifier that implies the whole system.

global log file

See **logglob**.

I

interesting process

A process becomes interesting when it is first created, when it ends, and when it exceeds user-defined thresholds for cpu use, disk use, response time, and so on.

L

log file set

A collection of files that contain data collected from one source.

logappl

The raw log file that contains measurements of the processes in each user-defined application.

logdev

The raw log file that contains measurements of individual device (such as disk and `netif`) performance.

logglob

The raw log file that contains measurements of the system-wide, or global, workload.

logindx

The raw log file that contains additional information required for accessing data in the other log files.

logproc

The raw log file that contains measurements of selected “interesting” processes. A process becomes interesting when it is first created, when it ends, and when it exceeds user-defined thresholds for CPU use, disk use, response time, and so on.

logtran

The raw log file that contains measurements of transaction data.

M

midaemon

An OV Performance Agent program that translates trace data into Measurement Interface counter data using a memory based MI Performance Database to hold the counters. This database is accessed by collector programs such as `scopeux`.

mwa script

The OV Performance Agent script that has options for starting, stopping and restarting OV Performance Agent processes such as the `scopeux` data collector, `midaemon`, `ttd`, `coda`, `ovc`, `ovbbccb`, `perflbd`, `rep_server`, and the alarm generator. See also the `mwa` man page.

O

ovbbccb

The OpenView Operations Communication Broker for HTTP(S) based communication controlled by `ovc`. See also **`coda`** and **`ovc`**.

ovc

The OpenView Operations controlling and monitoring process. In a standalone OVPA installation, `ovc` monitors and controls `coda` and `ovbbccb`. If OVPA is installed on a system with OpenView Operations for UNIX 8.x agent installed, `ovc` also monitors and controls OpenView Operations for UNIX 8.x processes. See also **`coda`** and **`ovbbccb`**.

ovpa script

The OV Performance Agent script that has options for starting, stopping and restarting OV Performance Agent processes such as the `scopeux` data collector, alarm generator, `ttd`, `midaemon`, `ovc`, `ovbbccb`, and `coda`. See also the `ovpa` man page.

OV Performance Manager

Provides integrated performance management for multi-vendor distributed networks. It uses a single workstation to monitor environment performance on networks that range in size from tens to thousands of nodes.

P

parm file

An OV Performance Agent file containing the parameters used by `scopeux` to customize data collection.

perflbd.rc

A configuration file residing in the `/var/opt/perf/` directory. This file is maintained as a symbolic link to the `datasources` file. See also **datasources file.**

perfstat

A program that displays the status of all performance processes in your system.

PerfView

See **OV Performance Manager**.

process

Execution of a program file. It can represent an interactive user (processes running at normal, nice, or real-time priorities) or an operating system processes.

process log file

See **logproc**.

R

raw log file

Summarized measurements of system data collected by `scopeux`. *See* **logappl**, **logproc**, **logdev**, **logtran**, and **logindx**.

real time

The actual time in which an event takes place.

repository server

A server that provides data to the alarm generator and the OV Performance Manager analysis product. There is one repository server for each data source configured in the `perflbd.rc` configuration file. *See also* **data source**.

resize

Changes the overall size of a log file using the `utility` program's `resize` command.

run file

Created by the `scopeux` collector to indicate that the `scopeux` process is running. Removing the `run` file causes `scopeux` to terminate.

S

scopeux

The OV Performance Agent data collector program that collects performance data and writes (logs) it to raw log files for later analysis or archiving. *See also* **raw log files**.

scopeux log files

See **raw log files**.

status.scope file

Created by the `scopeux` collector to record status, data inconsistencies, and errors.

system ID

The string of characters that identifies your system. The default is the host name as returned by `uname -n`.

T

transaction log file

See **logtran**.

transaction tracking

The technology used in OV Performance Agent that allows information technology (IT) resource managers to measure end-to-end response time of business application transactions.

ttd.conf

The transaction configuration file where you define each transaction and the information to be tracked, such as transaction name, performance distribution range, and service level objective.

U

utility

An OV Performance Agent program that allows you to open, resize, scan, and generate reports on raw and extracted log files. You can also use it to check `parm` file and `alarmdef` file syntax, and obtain alarm information from historical log file data

Index

R

S

T

U

V