

# HP OpenView Performance Agent

For HP-UX 11i v1 and Later

Software Version: C.04.50

---

## Installation and Configuration Guide

Manufacturing Part Number: B4967-90069  
October 2005



## Legal Notices

### Warranty

*Hewlett-Packard makes no warranty of any kind with regard to this document, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Hewlett-Packard shall not be held liable for errors contained herein or direct, indirect, special, incidental or consequential damages in connection with the furnishing, performance, or use of this material.*

A copy of the specific warranty terms applicable to your Hewlett-Packard product can be obtained from your local Sales and Service Office.

### Restricted Rights Legend

Use, duplication, or disclosure by the U.S. Government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause in DFARS 252.227-7013.

Hewlett-Packard Company  
United States of America

Rights for non-DOD U.S. Government Departments and Agencies are as set forth in FAR 52.227-19(c)(1,2).

### Copyright Notices

© Copyright 2005 Hewlett-Packard Development Company, L.P.

No part of this document may be copied, reproduced, or translated into another language without the prior written consent of Hewlett-Packard Company. The information contained in this material is subject to change without notice.

### 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

Please visit the HP OpenView support web site at:

**<http://www.hp.com/managementsoftware/support>**

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

HP OpenView online software 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 valuable support customer, you can benefit by using the support site to:

- Search for knowledge documents of interest
- Submit enhancement requests online
- Download software patches
- Submit and track progress on support cases
- Manage a support contract
- Look up HP support contacts
- Review information about available services
- Enter 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 log in. Many also require a support contract.

To find more information about access levels, go to:

**[http://www.hp.com/managementsoftware/access\\_level](http://www.hp.com/managementsoftware/access_level)**

To register for an HP Passport ID, go to:

**<http://www.managementsoftware.hp.com/passport-registration.html>**



# Contents

|          |   |           |
|----------|---|-----------|
| <b>1</b> | <b>Installing OV Performance Agent</b> .....            | <b>7</b>  |
|          | Introduction .....                                      | 7         |
|          | Installation Requirements.....                          | 9         |
|          | Hardware.....   | 9         |
|          | Software.....   | 9         |
|          | Communication Protocols.....                            | 9         |
|          | Disk Space.....   | 10        |
|          | Installation Procedures .....                           | 11        |
|          | Stopping Active Performance Tools or Processes .....    | 11        |
|          | Installing OV Performance Agent .....                   | 12        |
|          | Deploying OV Performance Agent Using OV Operations..... | 13        |
|          | Installing Files on a Remote Drive .....                | 13        |
|          | Removing OV Performance Agent.....                      | 15        |
| <b>2</b> | <b>Starting and Running OV Performance Agent</b> .....  | <b>17</b> |
|          | Introduction .....                                      | 17        |
|          | Starting and Stopping OV Performance Agent .....        | 18        |
|          | Changing Protocols .....                                | 21        |
|          | Starting and Stopping Automatically .....               | 22        |
|          | Status Checking .....                                   | 23        |
|          | Examples Directory .....                                | 23        |
|          | Communicating Across Firewall.....                      | 24        |
|          | Communicating in the HTTP Environment .....             | 25        |
|          | Communicating in the DCE Environment .....              | 30        |
|          | Configuring Secure Communication.....                   | 37        |
|          | Using Certificates .....                                | 37        |
|          | Using Client Authentication .....                       | 37        |
|          | Configuring Data Sources .....                          | 41        |

|   |    |
|---|----|
| Datasources Configuration File Format ..... | 42 |
| Parm File Configuration .....               | 43 |
| Defining Alarms .....                       | 44 |
| Native Language Support .....               | 45 |
| Viewing and Printing Documents .....        | 46 |
| Viewing Documents on the Web .....          | 47 |
| Adobe Acrobat Files .....                   | 47 |
| ASCII Text Files .....                      | 47 |
| Glossary .....                              | 49 |
| Index .....                                 | 57 |

---

# 1 Installing OV Performance Agent

## Introduction

HP OpenView Performance Agent software captures performance, resource, and transaction data from your HP 9000 server, Integrity server, or workstation. Using minimal system resources, the software continuously collects, logs, summarizes, and time stamps data, and detects alarm conditions on current and historical data across your system. You can analyze the data using spreadsheet programs, Hewlett-Packard analysis products such as OV Performance Manager, or third-party analysis products. Also, OV Performance Agent provides data access to OV Performance Manager, HP OpenView Network Node Manager, and HP OpenView Operations.



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 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.
- Perform service-level management based on transaction response time.
- Perform capacity planning.
- Respond to alarm conditions.
- Solve system management problems before they arise.

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



If you are running OV Performance Agent on a factory integrated system, skip this chapter and go directly to, [Chapter 2, Starting and Running OV Performance Agent](#).



# Installation Requirements

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



If you are planning to run a large number of `dsilog` processes, you may need to increase the values of the HP-UX kernel parameters `shmmni` and `nlocks`. `shmmni` specifies the maximum number of shared memory segments. `nlocks` specifies the maximum number of file locks on a system. The default value for each is 200.

Each active DSI log file set uses a shared memory segment and one or more file locks. You can change the settings for `shmmni` and `nlocks` through the System Administration and Maintenance utility (SAM).

## Hardware

OV Performance Agent runs on HP 9000 and Integrity Servers, and HP Workstations.

## Software

This version of OV Performance Agent requires releases 11i v1 or later of the HP-UX operating system.

## Communication Protocols

OV Performance Agent supports the following communication protocols:

- HTTP(S) 1.1
- DCE

## Disk Space

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

- For first time installation of OV Performance Agent, 100 MB of disk space is required in the `/opt/perf/` and `/opt/OV/` directories.
- For OVPA databases and status files, allow for 125 MB of disk space in the `/var/opt/perf/` and `/var/opt/OV/` directories.

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

# Installation Procedures

OV Performance Agent is on the CD-ROM installation media in several file sets. The size of the product is approximately 260 MB.

OV Performance Agent installation is done in two phases:

- 1 If you have previously installed OVPA or Glance on the system, stop any performance tools or processes that may be running. See the following section [Stopping Active Performance Tools or Processes](#).
- 2 Install OV Performance Agent. See [Installing OV Performance Agent](#) on page 12.

## Stopping Active Performance Tools or Processes

- 1 Log in as **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 Glance, stop them. (Before doing so, make sure that none of your users are currently using these tools.)
- 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` files 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 Make sure the `midaemon` (measurement interface daemon) is not active by typing:

```
ps -ef | grep midaemon
```

If the `midaemon` is still active, type:

```
/opt/perf/bin/midaemon -T
```

- 5 Make sure the `ttd` (transaction tracking daemon) is not active by typing:

```
ps -ef | grep ttd
```

If the `ttd` is still active, type:

```
/opt/perf/bin/ttd -k
```



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

- 6 Run `perfstat` again to make sure that no performance tools or processes are active. When all tools or processes have been stopped, proceed with the installation.

## Installing OV Performance Agent

- 1 Make sure you are logged in as **root**.
- 2 Run the `swinstall` program to install OV Performance Agent the same way you install any other software package on an HP-UX system.
- 3 The OV Performance Agent product is included as part of the GlancePlus Pak. Change the `swinstall` source to point to your installation media and select the GlancePlus Pak bundle,

or,

select the OV Performance Agent bundle if you have not purchased the GlancePlus Pak.

- 4 The install process appends information to the end of the two log files named `/var/adm/sw/swinstall.log` and `/var/adm/sw/swagent.log`. These files contain system messages about the installation and records of any problems the system encountered. Read these files to see if any error messages were logged during the installation and take appropriate action as needed.

To read the last 100 lines of the `swinstall.log` file, type the following command:

```
tail -100 /var/adm/sw/swinstall.log
```

If you run `swinstall` in an interactive mode, you can open the `swinstall.log` file and view it while `swinstall` runs. Otherwise, you can look at `swinstall.log` and `swagent.log` in `/var/adm/sw/` to see any errors or problems.

Installation of OV Performance Agent automatically updates files used by `/etc/profile/` and lets users access the product's executables in `/opt/perf/bin/` and its man pages without explicitly updating their shell environment variables. This should take effect the next time users log in.

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



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.

## Deploying OV Performance Agent Using OV Operations

If you are using OV Operations for UNIX 7.x and 8.x, you can install OV Performance Agent from the management server to an HP-UX managed node.



You cannot install this version of OV Performance Agent on Itanium systems that have OV Operations 8.x management server installations.

For installation instructions on OV Operations for UNIX versions 7.x, please see Chapter 6, “About OpenView Performance Agent for UNIX,” in the *HP OpenView Operations for UNIX Administrator's Reference Volume II*. For installation instructions on OV Operations for UNIX versions 8.x, please see Chapter 4, “HP OpenView Performance Agent,” in the *HP OpenView Operations Administrator's Reference*.

## Installing Files on a Remote Drive

Some users of OV Performance Agent have linked the `/var/opt/perf/` directory to a different file system because of disk space concerns. If your `/var/opt/perf/` directory is redirected to a non-local file system, such as a remote NFS mount point, then be aware of the following considerations:

- OV Performance Agent does not support remote NFS logging in a WAN environment. OV Performance Agent is designed to optimally use locally mounted log files.
- Remote NFS mounting of log files in a LAN environment could cause network overloading depending on data logging thresholds, which can negatively impact the performance of OV Performance Agent and other networked applications. Also, log file maintenance can take substantially longer due to the remote access, which can cause time-out errors in alarm processing.



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

# Removing OV Performance Agent

- 1 Make sure you are logged in as **root**.
- 2 Run the `swremove` program to uninstall OV Performance Agent the same way you uninstall any other software package on an HP-UX system.
- 3 The OV Performance Agent product is included as part of the GlancePlus Pak. Select the GlancePlus Pak bundle,  
or,  
select the OV Performance Agent bundle if you have not purchased the GlancePlus Pak.
- 4 The uninstall process appends information to the end of the two log files named `/var/adm/sw/swremove.log` and `/var/adm/sw/swagent.log`. These files contain system messages about the removal and records of any problems the system encountered. Read these files to see if any error messages were logged during the removal and take appropriate action as needed.

To read the last 100 lines of the `swremove.log` file, type the following command:

```
tail -100 /var/adm/sw/swremove.log
```

If you run `swremove` in an interactive mode, you can open the `swremove.log` file and view it while `swremove` runs. Otherwise, you can look at `swremove.log` and `swagent.log` in `/var/adm/sw/` to see any errors or problems after `swremove` has run.

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 explains the tasks involved in starting up and running OV Performance Agent after it has been installed on your HP-UX system.

The following topics are discussed:

- procedures for starting and stopping OV Performance Agent
- setting variables in the `/etc/rc.config.d/ovpa` file
- running OV Performance Agent
- communicating across a firewall
- `parm` file configuration for HP-UX systems
- configuring data sources
- configuring alarm definitions



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*.

OV Performance Agent supports the HTTP data communication mechanism, along with the legacy DCE mode. If you are installing OV Performance Agent for the first time, the default data communication mode is HTTP. If you are upgrading to OV Performance Agent 4.5, the previously used DCE data communication mode is used by default. If you want to move to the new HTTP mode of data communication, you can change the configuration after installation. OV Performance Agent supports HTTPS data communication only in an HP OpenView Operations 8.x environment.

# 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 data communication mode is enabled by default. For information on changing the data communication protocol, see [Changing Protocols](#) on page 21.

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



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

The table below lists the different services that can be started for the different protocols.

**Table 1 OVPA services started for different protocols.**

| <b>Services started for HTTP protocol</b> | <b>Services started for DCE protocol</b> |
|---|--|
| <code>scopeux</code>                      | <code>scopeux</code>                     |
| <code>coda</code>                         | <code>coda</code>                        |
| <code>perfalarm</code>                    | <code>perfalarm</code>                   |
| <code>midaemon</code>                     | <code>midaemon</code>                    |
| <code>ttd</code>                          | <code>ttd</code>                         |
| <code>ovc</code>                          | <code>ovc</code>                         |
| <code>ovbbccb</code>                      | <code>ovbbccb</code>                     |

**Table 1 OVPA services started for different protocols.**

| <b>Services started for HTTP protocol</b> | <b>Services started for DCE protocol</b> |
|---|--|
|   | 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 `ovbbcbb`. `ttd` should always be left running. If OpenView Operations agent is running on the system, `ovpa stop` will 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.

- `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` processes 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 alarm generator process to 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 other processes.

## Using the `mwa` script

To start OV Performance Agent and its processes using `mwa`:

- 1 Log in as **root**.
- 2 Start OV Performance Agent and its processes by typing:

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

The `mwa start` script starts OV Performance Agent and all its processes, including `scopeux` (data collector), `midaemon` (measurement interface daemon), `ttd` (transaction tracking daemon), `coda`, `ovc`, `ovbbcbb`, `perflbd`, `rep_server`, and the alarm generator. As the script executes, the names of some of the processes that are running and the files that are in use are 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. `mwa stop` does not stop the `coda` daemon if other products are using `coda`.
- OV Performance Agent can be reinitialized using the `mwa restart` option. Changes to configuration files will *not* take effect on your system unless the processes are restarted.
- `mwa restart server` causes the `coda` daemon 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 starts the alarm generator processes and rereads the `alarmdef` file. The HTTP based alarm generator, `perfalarm`, is enabled by default<sup>1</sup>.
- `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` cause the alarm generator processes to temporarily stop and then restart and reread the `alarmdef` file. This action does not disrupt any other processes.

## Changing Protocols

During first time installation of OV Performance Agent, the protocol selected is written to the `/etc/rc.config.d/ovpa` file as an addition to the environment variable `MWA_PROTOCOL` (for example, `MWA_PROTOCOL=http`).

### To switch to DCE mode

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

1. To enable the DCE based alarm generator, `alarmgen`, stop OV Performance Agent, rename the `perfalarm` executable to `perfalarm.old`, and restart OV Performance Agent 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/rc.config.d/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. The `mwa` script starts the HTTP, DCE, or NCS data communication components depending on the value set for `MWA_PROTOCOL`. If `MWA_PROTOCOL` is set to `http`, `ovpa` and `mwa` scripts start the same components.

- ▶ 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 automatically starting OV Performance Agent whenever the system restarts, and of stopping whenever the system shuts down is controlled by the OVPA startup and shutdown script, and the OVPA startup configuration file `/etc/rc.config.d/ovpa`.

The OVPA startup configuration file contains various environment variables that control the behavior of OV Performance Agent at system startup. This file is a source file for the OVPA system startup and shutdown script.

You can modify the following environment and shell variables to change the default behavior of OV Performance Agent:

- `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 startup
  - 1      Start OV Performance Agent at the system startup
- The `MWA_START_COMMAND` is a variable that specifies the script options that are used to start OV Performance Agent whenever your system reboots. Normally, the variable is set to `/opt/perf/bin/mwa start`.

- ▶ The OVPA startup configuration file is not overwritten when OV Performance Agent is patched or updated, so that any customizations you made are preserved.

## 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 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/config/` 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

In general, 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 client 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 37.

The HTTP based interface is flexible, because it can use proxies, requires fewer ports and is firewall friendly. The `perflbd` and `rep_server` daemons, and their dependency on the DCE subsystem can still be used.

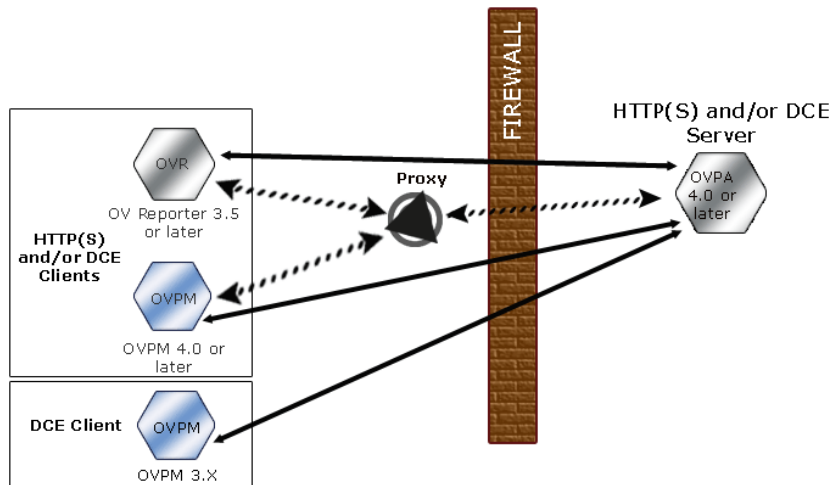
If a network firewall exists between two systems, OV Performance Manager and OV Reporter can get data from the OV Performance Agent system.

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

There are different ways to configure HTTP communication in a firewall environment. The recommended way is to use HTTP proxies for OVPA data communication 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.

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. 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.

Figure 1 on page 25 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 via the proxy.

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

## Ports Used for Communication

To access data collected by OVPA, ports for the HTTP server (OVPA) and the HTTP client (Reporter and OVPM) must be opened. There are two ways to configure HTTP clients in a firewall environment: with an HTTP proxy and without. In both cases, to access data from OVPA nodes, only one port needs to be opened on the HTTP server (OVPA) side.

### With HTTP Proxy

The recommended way is to use HTTP proxies when communicating through a firewall. This simplifies the configuration because proxies are often in use and the firewall needs to be opened only for the proxy system and for a smaller number of ports. It is recommended that you do not change the default 383 port.

Default Ports for OVPA (with proxies) are shown in Table 2.

**Table 2 OVPA Default Ports**

| Source | Destination | Protocol | Source Port          | Destination Port | Description          |
|--------|-------------|----------|----------------------|------------------|----------------------|
| PROXY  | MGD NODE    | HTTP     | Defined by the proxy | 383              | Communication Broker |

To configure the proxy, run 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.

For example:

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

### Without HTTP Proxy

If HTTP proxies are not available, additional configuration settings are required on the Reporter and OVPM system. See the section, [Configure Reporter/OVPM without HTTP Proxy](#) on page 29.

## Configuring Ports Across Firewall

To configure communications with OVPA in a firewall environment, follow these guidelines:

- Understand your firewall environment including the client and server data flow.
- Check the port usage in your environment.
- Refer to the following sections for details on configuring communication across firewalls, when proxies are used:
  - [Configure OVPA Ports](#)
  - [Configure Reporter and/or OVPM](#)
  - [Other Considerations](#)
- Test the communication across the firewall.

### Configure OVPA Ports

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

Configuring the Default Communication Broker Port:

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>
```

**ovc -restart**

Configuring OVPA for Single Port Communication:

On the OV Performance Agent system, coda uses a port that is dynamically allocated. To change the port setting to use the communication broker's port specified above, type the following commands:

```
ovconfchg -ns coda.commm -set SERVER_BIND_ADDR localhost  
ovc -restart
```

### Configure Reporter and/or OVPM

You can configure the HTTP clients (Reporter or OVPM for Windows or UNIX) in a firewall environment in one of two ways:

- With HTTP Proxy – This is the recommended way. See the section [Configure Reporter/OVPM with HTTP Proxy](#).
- Without HTTP Proxy – This is *not* the recommended way. See the section [Configure Reporter/OVPM without HTTP Proxy](#).

### Configure Reporter/OVPM with HTTP Proxy

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.

Configure OVPM 5.0 as follows:

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.

Configure Reporter/OVPM 4.x as follows:

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.**)

### **Configure Reporter/OVPM without HTTP Proxy**

If your firewall environment does not have proxies then you may want to specify the HTTP client ports directly if you want to filter based on both source and destination.

If Reporter and OVPM for Windows are installed on the same system and both access OVPA in parallel, you would specify a port range as described in this section. If they are running on different systems, you can instead specify a single port for each.

Configure OVPM 5.0 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/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
```

### **Other Considerations**

#### **About 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.

On the OVPA system, specify the `SERVER_BIND_ADDR` parameter.

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

On the OVPM 5.0 system, specify the `CLIENT_BIND_ADDR` parameter.

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

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:

- 1 Locate the lines that apply to `CLIENT_BIND_ADDR` and uncomment the line `;CLIENT_BIND_ADDR =`
- 2 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

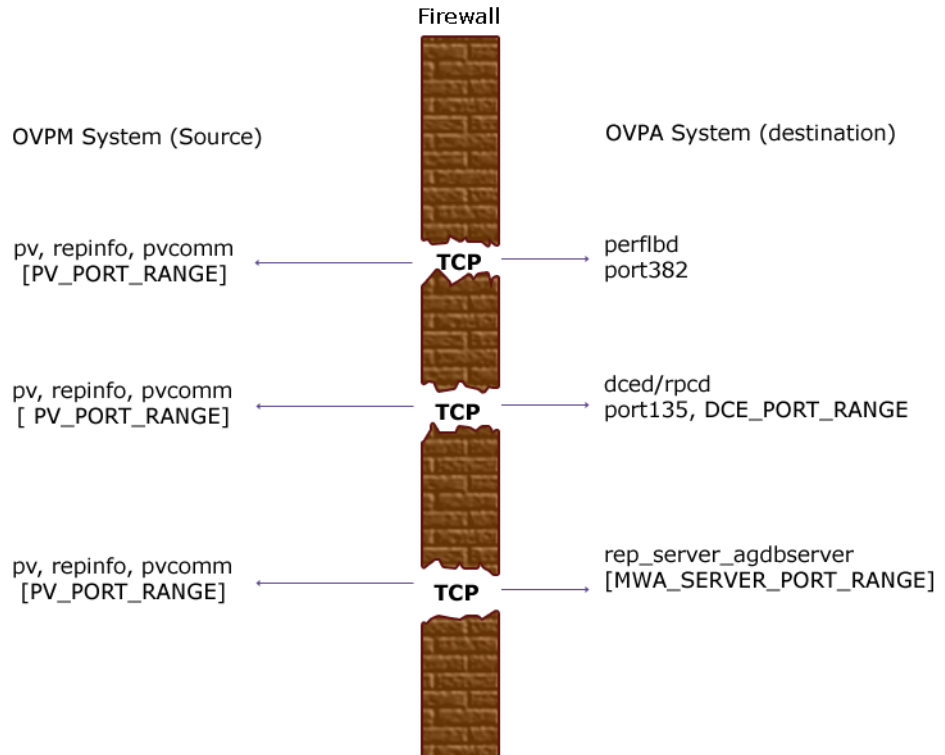
The method of configuring the socket port numbers for OV Performance Agent depends on the version of OV Performance Manager that will be communicating with OV Performance Agent. To configure the socket port numbers, follow the instructions in [Configuring OV Performance Manager C.03.00 and Later with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

#### 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:

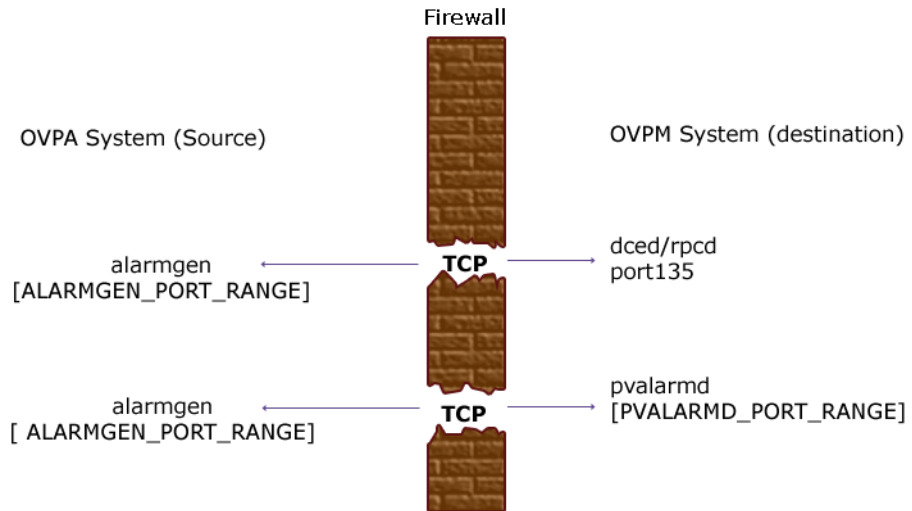
```
agdbserver    20001  
rep_server    20002  
rep_server    20003  
rep_server    20004
```

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-yyy]
```

where `xxxx-yyy` 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`.

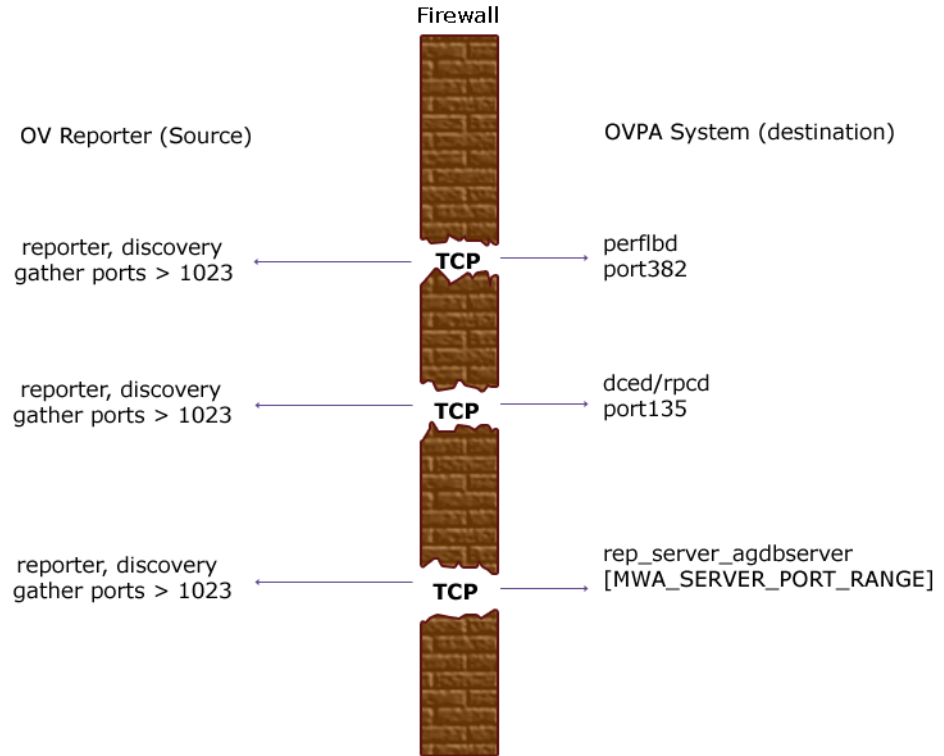
- ▶ Sending alarms from OV Performance Agent to OV Performance Manager through firewalls that use Network Address Translation (NAT) IP addresses is not supported.

## 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 30.

## 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 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
ovcodautl -config
```

## Using Client Authentication

OV Performance Agent enables optional authentication of client connections from products such as OV Performance Manger or OV 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.

## 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 pound sign (#) 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 42.

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:

```
/usr/lpp/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 Configuration

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 detailed information about defining alarms, see Chapter 7, “Performance Alarms,” in the *HP OpenView Performance Agent for UNIX User's Manual*. This chapter describes the alarm definitions syntax, how alarms work, and how alarms can be used to monitor performance.

# Native Language Support

A feature of the OV Performance Agent `extract` and `utility` programs running on HP-UX systems is native language support (NLS).

The `extract` and `utility` programs can support the date and time formats of other native languages if you use the NLS feature of the HP-UX operating system.

If a native language is installed and selected on your system, the `extract` and `utility` programs make the following adjustments:

- Dates and times are entered and printed in the language specified in the NLS package. If a date or time is entered in an unrecognizable format, the program prompts you with an example in the correct format.
- The text for the `help` command is obtained from `extract.help` and `utility.help`. These files are located in the `/opt/perf/help/ovpa/$LANG` directory where `$LANG` is the language specified for this system. The default language is “C.” If a help catalog cannot be found in the `$LANG` directory, the default help files are used.

## 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. The 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 OV Performance Agent Documentation Set**

| <b>Document</b>  | <b>File Name</b> | <b>UNIX Location</b>        |
|--|------------------|-----------------------------|
| <i>HP OpenView Performance Agent for HP-UX Installation &amp; Configuration Guide</i>      | ovpainst.pdf     | /opt/perf/paperdocs/ovpa/C/ |
| <i>HP OpenView Performance Agent for UNIX User's Manual</i>                                | ovpausers.pdf    | /opt/perf/paperdocs/ovpa/C/ |
| <i>HP OpenView Performance Agent for UNIX Data Source Integration Guide</i>                | ovpadsi.pdf      | /opt/perf/paperdocs/ovpa/C/ |
| <i>HP OpenView Performance Agent for UNIX &amp; GlancePlus: Tracking Your Transactions</i> | tyt.pdf          | /opt/perf/paperdocs/arm/C/  |
| <i>Application Response Measurement (ARM) API Guide</i>                                    | arm2api.pdf      | /opt/perf/paperdocs/arm/C/  |
| <i>HP OpenView Performance Agent HP-UX Metric Definitions</i>                              | methp.txt        | /opt/perf/paperdocs/ovpa/C/ |
| <i>OVPA metrics list by Data Class for all operating systems</i>                           | 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](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.

From HP-UX, you can read a .PDF file using the `acroread` command, if you have installed 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 Metrics.txt
```





---

# Glossary

This glossary contains an alphabetized list of some of the terms associated with OV Performance Agent.

## **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 OVPM) and to OVO. 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`.*

### **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.

### **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.

**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, OVPM. This file format is suitable for input to the `extract` and `utility` programs and is the preferred method for archiving performance data.

**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.*

**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.

**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) 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.

**midaemon**

An OV Performance Agent program that collects and counts trace data coming from the operating system kernel and stores the counters in the shared memory segment. This data is the main source of performance data for scopeux.

**ovbbccb**

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

**ovc**

The OpenView Operations controlling and monitoring process. In a standalone OVPA installation, ovcd monitors and controls coda and ovbbccb. If OVPA is installed on a system with OpenView Operations for UNIX 8.x agent installed, ovcd 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, `ttc`, `midaemon`, and `coda`. See also the `ovpa man` page.

### **OV Performance Manager (OVPM)**

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.

### **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**.

### **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 OVPM analysis product. There is one repository server for each data source configured in the `perflbd.rc` configuration file. See also **data source**.

**resize**

Changing 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.

**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 file.*

**scopeux log files**

*See raw log file.*

**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`.

**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.

**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

## A

- alarmdef file, 44
- alarms, 44
- authip file, 38
  - examples, 39
  - formatting, 39

## C

- changing protocols, 21
- client, firewall, 25
- client authentication, 38
- coda.log file, 42
- communicating across a firewall, 30
- configuring
  - data sources, 41
  - OVPM and OVPA firewall communication, 30
  - OVPM C.03.00 and later with OVPA firewall communication, 30
  - OV Reporter and OVPA firewall communication, 34
  - parm file, 43

## D

- data sources
  - configuring, 41
  - deleting, 41
  - DSI, 41
  - SCOPE, 41
  - scopeux, 41
- defining alarms, 44
- deleting data sources, 41
- disk space requirements, 10
- documents
  - OVPA documentation set, 46
  - viewing on the web, 47
  - viewing PDF files from UNIX, 47
- DSI data sources, 41

## E

- environment variables, 22
- examples directory
  - README, 23
- extract program, 42

## F

### files

- alarmdef, 44
- coda.log, 42
- OVPA startup configuration, 22
- parm, 43
- perflbd.rc, 41
- status.scope, 23

### firewall

- client, 25
- communicating across, 24
- configure OVPA server port, 27
- configure Reporter/OVPM without HTTP proxy, 29
- default ports for communication with OVPA, 26
- overview of OVPA communications configuration, 27
- server, 25
- systems with multiple IP addresses, 29

### firewalls, 30

- communicating across, 30
- configuring OV Reporter and OVPA communication, 34
- configuring with OVPM C.03.00 and later, 30
- configuring with OV Reporter A.02.00 and later, 35

## H

### hardware requirements, 9

### HTTP

- client, 26
- clients, 26
- proxies, 25

## I

### installation procedures, 11

- installing files on a remote drive, 13
- installing from CD-ROM, 12
- installing OVPA using OVO, 13

### installation requirements, 9

- disk space, 10
- hardware, 9
- software, 9

## M

### metric definitions, printing, 46

### midaemon, 11

### MWA\_START\_COMMAND variable, 22

### MWA\_START variable, 22

### mwa scripts

- mwa restart, 21
- mwa restart alarm, 21
- mwa restart server, 21
- mwa start, 20
- mwa stop, 21

## N

### native language support, 45

## O

### OV Operations

- using to install OVPA, 13

### OVPA

- removing, 15
- startup configuration file, 22
- status files, 23
- system startup and shutdown script, 22

### ovpa restart script, 20

### ovpa script, 19

- ovpa scripts
  - ovpa start, 19
- OV Performance Agent
  - alarms, 44
  - data collection, 43
  - disk space requirements, 10
  - documentation set, 46
  - installation requirements, 9
  - installing files on a remote drive, 14
  - installing using OVO, 13
  - restarting, 19, 20
  - startup, 17
  - stopping and restarting, 19
- OV Performance Manager
  - configuring OVPM C.03.00 and later
    - with OVPA firewall communication, 30
  - firewall configuration with OVPA, 30
- OVPM
  - configure without HTTP proxy for firewall communication, 29
- OV Reporter
  - firewall configuration, 34

## P

- parm file, 43
- perflbd.rc file, 41
- perfstat command, 11
- printable files, 46
- printing documents, 46
- proxies, HTTP, 25

## R

- removing OVPA, 15

- Reporter
  - configure with HTTP proxy for firewall communication, 29
  - see OV Reporter, 34

- restarting
  - OV Performance Agent, 19, 20

- restricting RPC addresses, 36

- RPC\_SUPPORTED\_NETADDRS
  - environment variable, 36

## S

- SCOPE default data source, 41

- scopeux
  - data sources, 41

- scripts
  - opva restart alarm, 20
  - opva stop, 20
  - OVPA system startup and shutdown, 22

- server, firewall, 25

- software requirements, 9

- starting
  - OV Performance Agent, 17, 18
  - using mwa, 20
  - using ovpa, 19

- status files, 23
  - coda.txt, 23
  - status.alarmgen, 23
  - status.mi, 23
  - status.perfalarm, 23
  - status.perflbd, 23
  - status.rep\_server, 23
  - status.scope, 23
  - status.ttd, 23

- stopping
  - OV Performance Agent, 19
  - processes prior to installation, 11

## T

ttd, 12

## V

variables, 22

- environment, 22

- MWA\_START, 22

- MWA\_START\_COMMAND, 22

viewing

- documents on the web, 47

- PDF documents from UNIX, 47