

# HP OpenView Performance Agent

For the IBM AIX Operating System

Software Version: 4.60

---

## Installation and Configuration Guide

Manufacturing Part Number: B7491-90081

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

© Copyright 1983-2006 Hewlett-Packard Development Company, L.P.

### 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](http://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](http://www.hp.com/managementsoftware/access_level)**

To register for an HP Passport ID, go to:

**[www.managementsoftware.hp.com/passport-registration.html](http://www.managementsoftware.hp.com/passport-registration.html)**



# Contents

<b>1</b>	<b>Installing or Upgrading OpenView Performance Agent</b>	<b>7</b>
	Introducing OV Performance Agent	7
	Installation Requirements	9
	Hardware	9
	Software	9
	Communication Protocols	9
	Disk Space	10
	Install or Upgrade Procedures	11
	Upgrade	11
	First Time Installation	11
	Install CD-ROM Contents	11
	Read Before Installation	13
	Stop Active Performance Tools or Processes	15
	Install OV Performance Agent	16
	Installing OV Performance Agent with OV Operations Agent Installed on Your System	18
	Deploying OV Performance Agent Using OV Operations	19
	The install.ovpa Script	20
	Removing OV Performance Agent	21
	OV Performance Agent on a Virtualized Environment	22
<b>2</b>	<b>Starting and Running OV Performance Agent</b>	<b>23</b>
	Introduction	23
	Starting and Stopping OV Performance Agent	24
	Using the ovpa script	25
	Using the mwa script	26
	Changing Protocols	28
	Starting OV Performance Agent Automatically	29

The /etc/default/ovpa File .....	29
Status Checking .....	32
Examples Directory .....	32
Communicating Across a Firewall .....	33
Communicating in the HTTP Environment .....	34
Configuring Systems with Multiple IP Addresses .....	39
Communicating in the DCE Environment .....	39
Configuring Secure Communication .....	47
Using Certificates .....	47
Using Client Authentication .....	47
Configuring OV Performance Agent to Run on a Cluster Node .....	51
Naming Scheme for IP Addresses .....	51
Configuring Data Sources .....	58
Datasources Configuration File Format .....	59
Parm File .....	60
Defining Alarms .....	61
Viewing and Printing Documents .....	62
<b>A Configuring Coda .....</b>	<b>65</b>
Introduction .....	65
<b>Glossary .....</b>	<b>69</b>
<b>Index .....</b>	<b>77</b>

# 1 Installing or Upgrading OpenView Performance Agent

## Introducing OV Performance Agent

HP OpenView Performance Agent (OVPA) captures performance, resource, and transaction data from your IBM AIX 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).

HP OpenView Performance Agent supports monitoring of virtualized environments like LPARs. A new class of metrics BYLS is introduced to capture performance and resource data for the LPARs.

OV Performance supports monitoring of LPARs. For more information, refer to the section, [OV Performance Agent on a Virtualized Environment](#).



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:

- IBM RS/6000 and pSeries Systems

## Software

- OV Performance Agent requires IBM AIX 5L V5.1, AIX 5L V5.2 or AIX 5L V5.3 ML3 and later.
- The `libc.a` library is required for the OV Performance Agent to operate properly. The library is bundled within the `xlC.rte` package, available from your AIX OS CD-ROM disk media.
- The `libSpmi.a` library is a prerequisite for the memory metrics to be calculated accurately and also to collect cross-partition metrics for the BYLS class. The library is bundled within the `perfagent.tools` file set from your AIX OS CD-ROM disk media and is installed in the `/usr/lib/` directory.
- BYLS class of metrics is supported only on IBM AIX 5L V5.3 ML3 and later.
- To collect and log cross-partition metrics, either `xmservd` or `xmtopas` should be available. `xmtopas` is a part of `perfagent.tools` fileset and `xmservd` is a bundled software of the Performance Toolbox for AIX component.

## Communication Protocols

OV Performance Agent supports the following communication protocols:

- HTTP(S) 1.1
- NCS 1.5.1
- DCE V3.2

## Disk Space

OpenView Performance Agent installs in the `/usr/lpp/perf/` and `/usr/lpp/OV/` 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 `/usr/lpp/perf/` and `/usr/lpp/OV/` directories.
- For OVPA databases and status files, allow for 125 MB of disk space in the `/var/opt/OV/` and `/var/opt/perf/` directories.

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.

OV Performance Agent can run in HTTP, native NCS, and native DCE mode, as well as in emulated NCS mode via DCE, depending on the communication protocol and fileset selected.

## Upgrade

If you are upgrading OV Performance Agent, the previously used DCE or NCS protocol is retained by default. If you have previously installed OVPA or GlancePlus on the system, you must perform the following tasks:

- Create directory mounts to redirect the base directories to other locations in the filesystem. For more information, refer to the section [Read Before Installation](#) on page 13.
- Stop any performance tools or processes that may be running. For instructions, refer to the section [Stop Active Performance Tools or Processes](#) on page 15.
- Install OVPA. For instructions to install OpenView Performance Agent, refer to the section [Install OV Performance Agent](#) on page 16.

## First Time Installation

If you are installing OV Performance Agent for the first time, by default, the data communication protocol is set to HTTP. For step to install OVPA, refer to the section [Install OV Performance Agent](#) on page 16.

## Install CD-ROM Contents

The CD-ROM contains three installation filesets in tar archive files:

- `tarfile` consists of common OV Performance Agent files.

This fileset is needed regardless of the communication protocol used by OV Performance Agent. The fileset includes files required to run in HTTP mode.

- `tarncs` consists of files specific for OV Performance Agent using `llbd` to run in NCS mode.
- `tardce` consists of files specific for OV Performance Agent using `dced` to run in:
  - DCE mode
  - emulated NCS mode using the libraries `libdce.a`, `libdcephreads.a` and `libdcelibc_r.a`

Files from `tarfile` are always installed, as well as files from either `tarncs` or `tardce`, depending on which type of installation you choose.

Depending on which fileset is installed on the system, `perfstat -v` output lists major OV Performance Agent executable components along with the appropriate suffix which is either NCS or DCE.

The following example is an excerpt from output of `perfstat -v` and shows that NCS version of `scopeux` and `perflbd` are installed, as well as a common `ttd` executable:

```
scopeux  C.04.60.00 NCS 09/11/06 AIX 5.1+
ttd      C.04.60.00      09/11/06 AIX 5.1+
perflbd  C.04.60.00 NCS 09/11/06 AIX 5.1+
```

When running the installation script, you can choose a protocol and the fileset to install. For more information on installation options, see [The `install.ovpa` Script](#) on page 20.

To install the DCE fileset, you must have the DCE Runtime Services installed. Check for the correct packages by running `lsllpp -L | grep dce` to verify that the `dce.client` filesets are installed. If the filesets are *not* installed, install them, as they are a prerequisite for OV Performance Agent to run properly. If `dced` was *not* already running before installation, you have to start the `dced` daemon and OV Performance Agent manually. For more information on how to install DCE refer to your DCE-specific documentation. You can start OV Performance Agent using the `mwa` script.

By installing the DCE fileset, you have the capability of `dced` daemon to use either the DCE communication protocol or to emulate the NCS communication protocol

The NCS fileset installation uses the llbd (Local Location Broker Daemon) for NCS communication.



The dced and llbd daemons cannot run at the same time. Make sure the dced daemon is *not* already running on your system by using `ps -ef | grep dced`. If you are sure no other programs are using the dced, you can use `/usr/bin/stop.dce` to stop it. Refer to [Starting OV Performance Agent Automatically](#) on page 29 in Chapter 2 for information about starting the daemons automatically.

If you have HP OpenView Operations agent installed on your system, see [Installing OV Performance Agent with OV Operations Agent Installed on Your System](#) on page 18.

During the process of installation a `/var/opt/perf/.ovpa_binaryset` file is created and the information about which fileset has been installed is written to it. It contains the text `Selected Binary= tarncs` if NCS fileset is installed, or the text `Selected Binary= tardce` if DCE fileset is installed. This information is used by the installation and removal scripts. The `ovpa_binaryset` file should *not* be edited manually.

## Read Before Installation

This section provides information that you should consider, if you have used base directory symbolic links, to redirect the install location to different directories. Else you may proceed to with the installation instructions described in the subsequent sections.

On AIX platform, you can create base directory symbolic links, to redirect the install location to a different directory. The base directory symbolic links will be preserved during the first time installation of the product.

However, the symbolic links may not work when you are upgrading to a newer version of OVPA. When you are upgrading from a older version, as a part of the upgrade process, `installp` removes the symbolic links and replaces it with the default directory.

Hence the use of symbolic links to redirect base directories is not recommended and not supported on AIX.



You may face the same problem if you install any other OpenView product on your system. (Example: OVPA installed with the OVO agent)

Instead of symbolic links in base directories (`/usr/lpp/OV` and `/var/opt/OV` for OVO and additionally `/usr/lpp/perf` and `/var/opt/perf` for OVPA), use directory mounts to redirect the base directories to other locations in the filesystem.

The steps to redirect `InstallDir` and `DataDir` to a different file system are as follows:.



You must use directory mounts to redirect the base directories to other locations in the filesystem before the product installation.

The following steps are migrate an already existing installation from symbolic links to the usage of mount points:

- 1 Stop the OV software by typing:

```
ovc -kill
```

- 2 Remove the symbolic links

Perform the following tasks to mount the directories of the installed software at the new mount points.

- 3 Create mount points for `InstallDir` and `DataDir`

```
a mkdir /usr/lpp/OV
```

```
b mkdir /var/opt/OV
```

- 4 Add the mount points to `/etc/filesystems`

- 5 Edit `/etc/filesystems` and add entries for `InstallDir` and `DataDir` (filesystem type `namefs`)

### Example

To redirect `InstallDir` to `/mgmt/install/OV` and `DataDir` to `mgmt/data/OV`:

- 1 Open the file using the command

```
vi /etc/filesystems
```

- 2 Add the following entries

```
/usr/lpp/OV:
```

```
dev          = /mgmt/install/OV
vfs          = namefs
mount        = true
```

```

options          = rw
account          = false
/var/lpp/OV:

dev              = /mgmt/data/OV
vfs              = namefs
mount            = true
options          = rw
account          = false

```

- 3 Mount the filesystems by typing:

**a mount /usr/lpp/OV**

**b mount /var/lpp/OV**

- 4 Type the following command to verify the setup:

**mount**

The base directories InstallDir is redirected to /mgmt/install/OV and DataDir is redirected to mgmt/data/OV.

## Stop Active Performance Tools or Processes

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

**/usr/lpp/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, stop it by typing:

**/usr/lpp/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 /usr/lpp/perf/newconfig directory.

- 4 As a precaution, make sure you have backed up your customized configuration files such as the `parm`, `alarmdef`, `ttd.conf`, and `perflbd.rc` files, and any customized export template files.

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

- 5 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.

## Install OV Performance Agent

While installing OV Performance Agent, you can specify the data communication protocol to be used and the fileset to be installed.

If you are installing OV Performance Agent for the first time, by default, the data communication protocol is set to HTTP and the set of files from `tarncs` is installed if no additional option is specified at installation time. If you are upgrading OV Performance Agent to the current version, the previously used protocol and corresponding fileset are retained. For a detailed description of `install.ovpa` options, see [The `install.ovpa` Script](#) on page 20.

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

The table below lists the protocol used and the fileset installed 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
<b>Protocol</b>			
<code>-p http</code>	HTTP	HTTP	HTTP
<code>-p dce</code>	DCE	DCE	NCS
<code>-p ncs</code>	NCS	DCE	NCS



**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
No protocol specified	HTTP in a first time installation of OVPA and the previously used protocol on upgrade to the current version of OVPA		
Fileset			
-b dce	tarfile, tardce	tarfile, tardce	tarfile, tarncs
-b ncs	tarfile, tarncs	tarfile, tardce	tarfile, tarncs
No fileset specified	tarfile and tarncs in a first time installation of OVPA and, tarfile and the previously installed fileset on upgrade to the current version of OVPA		

To install:

- 1 Make sure you are logged in as user **root**.
- 2 Mount the CD-ROM to a file system (using **SMIT** or the **mount** command).
- 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. The directory contains the **install.ovpa** script.
- 5 Run the installation script.

To install using the HTTP communication protocol, type:

```
./install.ovpa -p http -b ncs
```

To install using the DCE communication protocol, type:

```
./install.ovpa -p dce -b dce
```

To install using the NCS communication protocol in the emulated mode, type:

```
./install.ovpa -p ncs -b dce
```

To install using the NCS communication protocol, type:

```
./install.ovpa -p ncs -b ncs
```

For details on changing the communication protocol after installation, see [Changing Protocols](#) on page 28 in Chapter 2.

The installation script automatically starts all OV Performance Agent processes. If you do *not* want OV Performance Agent to start after installation, run the installation script with the option `-R`.



If you have HP OpenView Operations agent installed on your system, see [Installing OV Performance Agent with OV Operations Agent Installed on Your System](#) on page 18.

The OV Performance Agent processes are also started or stopped automatically if you restart or shutdown. See [Chapter 2, Starting and Running OV Performance Agent](#).

6 Exit the CD-ROM directory by typing:

```
cd /
```

using `SMIT` or the `umount` command.

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.



If you are also running the GlancePlus product on your system, be sure to update GlancePlus to the same release version as OV Performance Agent. Both OV Performance Agent and GlancePlus 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 whether the OV Operations Agent 7.x is installed on your system and which communication protocol it is using. This information is gathered by checking for the existence and reading the contents of the `/var/opt/OV/conf/OpC/nodeinfo` file.

While upgrading OV Performance Agent, if OVO Agent 7.x is found, the `install.ovpa` script overrides any options you may have specified and notifies you of the options that will be used for installation:

- If NCS service is detected, the `install.ovpa` script is started with the `-p ncs -b ncs` options, enforcing the deployment of the NCS communication protocol and the NCS set of files. The following message is displayed:

```
OVO or OVO subagent has been found on your system.  
Installation will continue with -p ncs -b ncs option.
```

- If DCE service is detected, the `install.ovpa` script is started with the `-p dce -b dce` options, enforcing the deployment of the DCE communication protocol and the DCE set of files. The following message is displayed:

```
OVO or OVO subagent has been found on your system.  
Installation will continue with -p dce -b dce option.
```

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, the HTTP communication protocol and the NCS set of files are installed by default. If you are upgrading OV Performance Agent, the existing communication protocol is used, and the corresponding set of files is installed. For more information on how OV Performance Agent is installed, see [Install OV Performance Agent](#) on page 16.

## Deploying OV Performance Agent Using OV Operations

If you are using HP OpenView Operations for UNIX 8.x, you can install HP OpenView Performance Agent from the management server to an IBM AIX managed node.

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 [-hR] [-p dce | ncs | http] [-b dce | ncs]
```

The command line options have the following meaning:

- h        Display this message and exit.
- R        Do *not* start OV Performance Agent upon successful installation. By default, OV Performance Agent is automatically started.
- p        Enforce the deployment of the selected communication protocol.
  - http      Use HTTP communication protocol
  - dce        Use DCE communication protocol
  - ncs        Use NCS communication protocol
- b        Enforce the installation of the selected fileset.
  - dce        Install the DCE fileset
  - ncs        Install the NCS fileset

When no options are specified, and you are installing OV Performance Agent for the first time, the default is to install the NCS fileset and use the HTTP communication protocol. The option `-p dce -b ncs` is *not* allowed, since the NCS fileset does *not* support the DCE communication protocol.

# Removing OV Performance Agent

If you need to remove OV Performance Agent from a system, use the `ovpa.remove` script that is in the `/usr/lpp/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 `/usr/lpp/perf/newconfig` directory.

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.

# OV Performance Agent on a Virtualized Environment

OV Performance Agent installed on the LPARs provides a CEC (Central Electronics Complex) wide view. OVPA uses the RSI (Remote Statistics Interface) interface to discover all the LPARs configured on a CEC and to collect performance (the BYLS class of metrics) data from the LPARs. The list of LPARs discovered using RSI interface is controlled through configuration file `Rsi.hosts`. The RSI interface searches for the configuration file in following directories (in the listed order): `$HOME`, `/etc/perf` and `/usr/lpp/perfmgr`.



BYLS class of metrics is supported only on IBM AIX 5L V5.3 ML3 and later.

If any of the LPARs is restricted from responding through the configuration file `Rsi.hosts` then information about those LPARs will not be available. For information on `Rsi.hosts`, refer to the latest IBM documentation on Remote Statistics Interface Programming.



OVPA might not always discover all the LPARs configured on the current CEC.

If the configuration file `Rsi.hosts` is not available, then the RSI interface sends invitational broadcast messages to all the systems within the network. From the list of systems which respond to the message, OVPA discovers LPARs configured on the CEC.



Some LPARs may not be reported by OVPA if they are not responding to RSI calls within the `timeout` period. This `timeout` period cannot be configured from OVPA.

## 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 IBM RS/6000 system. The following topics are discussed:

- [Starting and Stopping OV Performance Agent](#)
- [Communicating Across a Firewall](#)
- [Configuring Secure Communication](#)
- [Configuring OV Performance Agent to Run on a Cluster Node](#)
- [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, `mwa` and `ovpa`, 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 28.

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:

```
/usr/lpp/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: **/usr/lpp/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 process.

## Using the mwa script

To start OV Performance Agent and its processes:

1 Log in as user **root**.

2 Type:

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



You can start OV Performance Agent using DCE communication protocol only if you have the DCE set of binaries installed.

If you are running OV Performance Agent supporting a DCE communication protocol and you want to change to NCS (NCS files must be installed on your system), you have to edit the `/etc/default/ovpa` file and set the `MWA_PROTOCOL` variable to `ncs` and the `MWA_LLBD_COMMAND` to `/usr/lpp/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 stop the `dced` daemon before running the `./mwa start` script. To confirm that `llbd` is running, run:

```
ps -ef | grep llbd
```

If OV Performance Agent is started in the normal boot sequence, all client applications using NCS (`llbd`) must be started after the startup of all HP OpenView products (OV Performance Manager, OV Operation, OV Performance Agent, 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), `ttdd` (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 `ttdd` (transaction tracking daemon), `ovc`, and `ovbbccb`. `ttdd` 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 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 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` cause 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 29, 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`, 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 or fileset you have selected for installation. The daemons used for HTTP data communication are always installed and active on your system.

## Starting OV Performance Agent 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/inittab`.

If you do *not* want OV Performance Agent to start automatically, remove the line that begins with `mwa` in the file `/etc/inittab` or set the variable `MWA_START =0` in the `/etc/default/ovpa` file.

After rebooting the system, the `dced` daemon has to be started prior to OV Performance Agent. To ensure the proper startup sequence you can edit the `/etc/inittab` file. For example, insert the following line in front of the line that begins with `mwa` in the `/etc/inittab` file:

```
rcdce:2:wait:/etc/rc.dce core > /dev/console 2 > &1
```



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:

- `/usr/lpp/perf/bin/ovpa` OV Performance Agent control script
- `/etc/rc.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 `/usr/lpp/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 changed to one of the following values:

<code>http</code>	run OV Performance Agent as an HTTP service
<code>ncs</code>	run OV Performance Agent as an NCS service
<code>dce</code>	run OV Performance Agent as a DCE service (only possible if the DCE set of binaries is installed)

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 `/usr/lpp/perf/bin/mwa start`.
- The `MWA_LLBD_COMMAND` contains the command string to start the NCS local location broker daemon. The `dcled` 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="/usr/lpp/perf/bin/llbd"`**



The variable only applies when OV Performance Agent operates in NCS mode if `llbd` or `dcled` 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
```

- `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-yyy]
:ncacn_ip_tcp [xxxx-yyy]
export RPC_RESTRICTED_PORTS
```

For more information about Firewall support, see [Communicating Across a Firewall](#) on page 33.

- 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_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.
- 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 and 127.0.0.2 IP addresses from the example below with a list of real addresses, separated by a colon (:).

```
RPC_SUPPORTED_NETADDRS=127.0.0.1:127.0.0.2
export RPC_SUPPORTED_NETADDRS
```

## 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/perf/status.ls  
/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 `/usr/lpp/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 `/example/ovpaconfig/` subdirectory contains sample alarm definitions and examples of `parm` file application-specific parameters. For more information, see the `/usr/lpp/perf/examples/README` file.



# Communicating Across a 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.



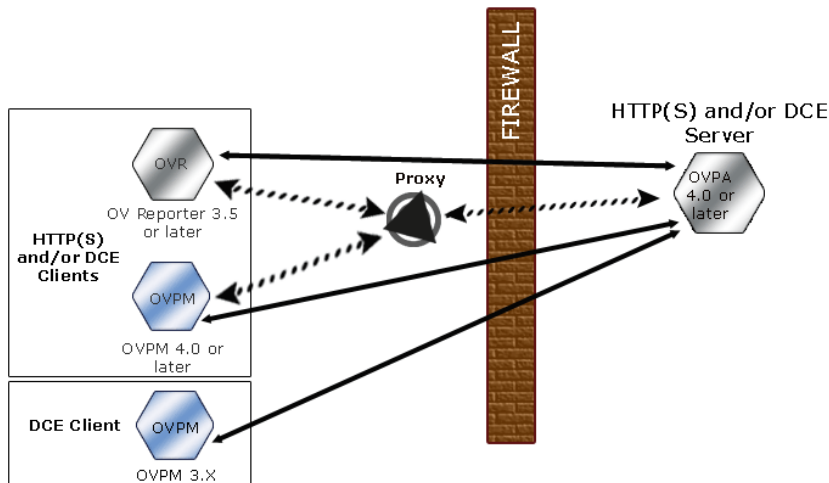
OV Performance Agent supports certificate-based secure (HTTPS) data communication only in the HP OpenView Operations 8.x environment. For more information, see [Using Certificates](#) on page 47.

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 34 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 daemon 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.



This section is applicable only if you are using the DCE communication protocol. NCS communication protocol uses different port ranges and cannot be used with a firewall.

## 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 40.

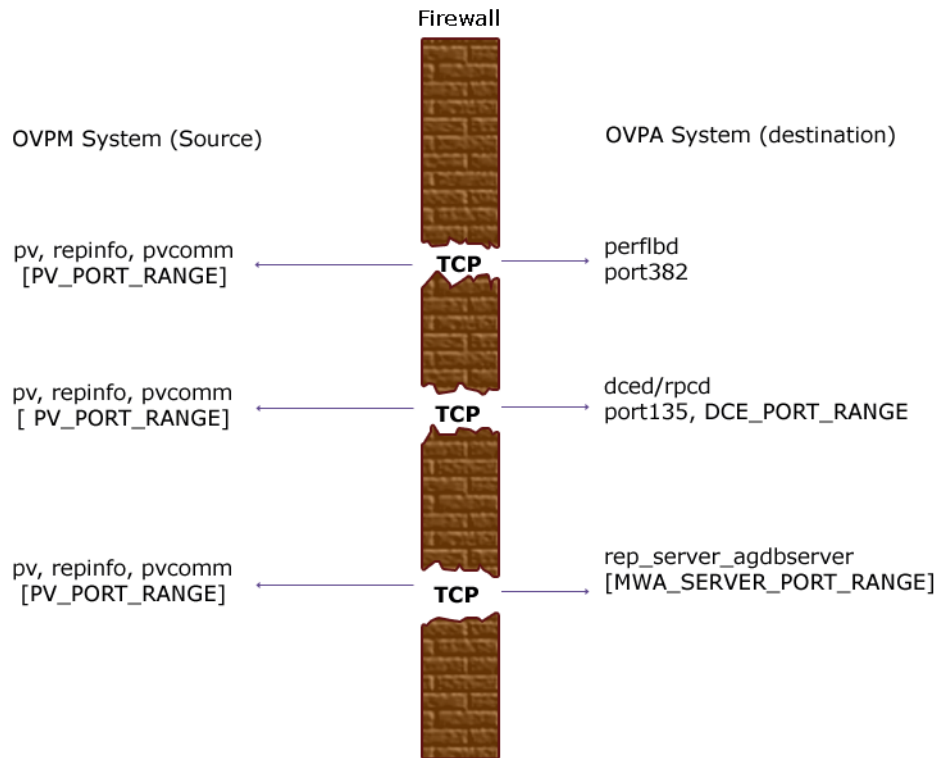
### [Configuring OV Performance Manager C.03.00 and Later with OV Performance Agent Communication](#)

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
```

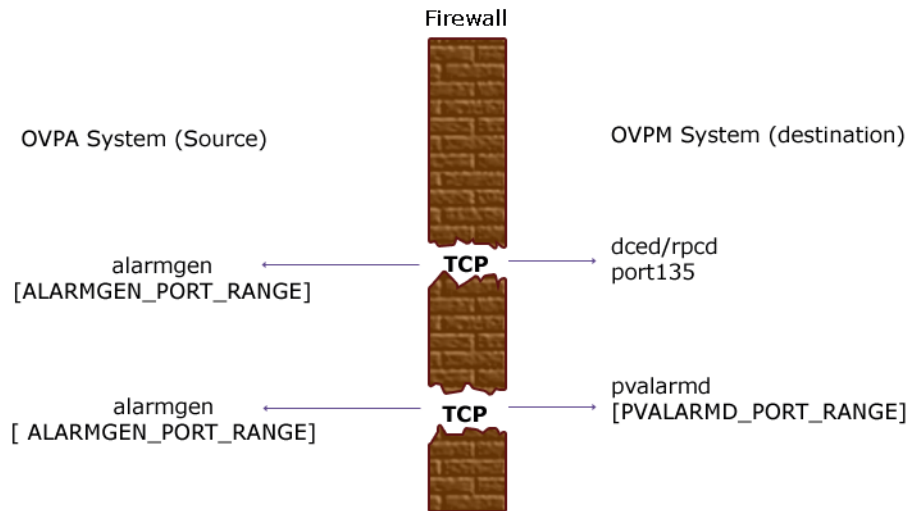
and there are three data sources configured in the `perflbd.rc` file, OV Performance Agent will use the following TCP port numbers:

```
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 `/usr/lpp/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`.



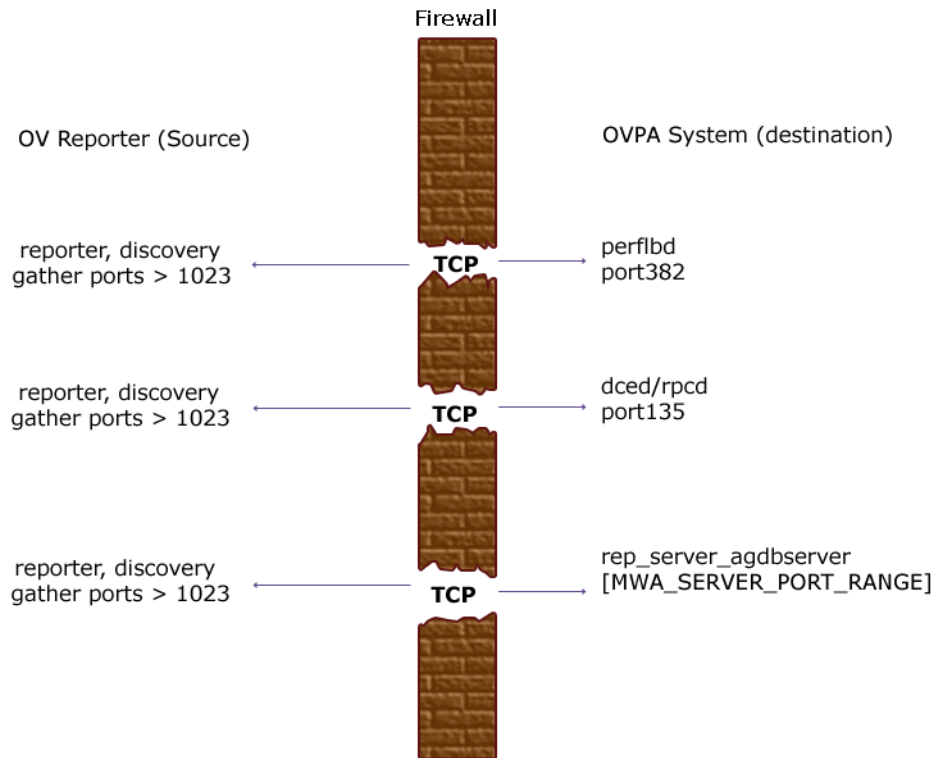
Sending alarms from OV Performance Agent to OV Performance Manager through firewalls that use Network Address Translation (NAT) IP addresses is not supported.

## 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 40.

## 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_NETADDRES` environment variable before starting the servers.

The syntax is:

```
RPC_SUPPORTED_NETADDRES=protocol:ip_address[,protocol:ip_addresses]
```

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

```
RPC_SUPPORTED_NETADDRES=ncadg_udp:192.1.1.1  
export RPC_SUPPORTED_NETADDRES
```

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.

```
/usr/lpp/perf/bin/mwa stop server
```

- 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/ovpa`.

Uncomment the following lines:

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

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

- Start OV Performance Agent, run:

```
/usr/lpp/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  
ovcodutil -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 OV Performance Agent to Run on a Cluster Node

OV Performance Agent running in DCE mode can run on a cluster node, that is on systems running HACMP for AIX software. In order to run on a cluster node OV Performance Agent has to have a fixed, always accessible IP address available.

Each cluster node system has its own OV Performance Agent. Each system must be accessible on a fixed IP address, which represents the system. The fixed, always accessible IP address is used by OV Performance Agent.

- If on the cluster node OVO agent is installed and running, use the OVO agent's IP alias (<systemname>\_ito). For more information refer to the *HP OpenView Operations for UNIX Administrator's Reference Volume II*. Proceed with [Configuring the OV Performance Agent Environment](#) on page 52.
- If an additional adapter (that is, network interface card) with a fixed IP address that is not used by HACMP (as a boot, service, or standby adapter) is available on a HACMP node, you can use this adapter's IP as the fixed, always accessible IP address used by OV Performance Agent. Proceed with [Configuring the OV Performance Agent Environment](#) on page 52.
- If none of the above is available on your cluster node you will have to assign each node an IP alias in the same network in which the boot and service IP addresses reside. In addition, you must configure the node in such a way that this IP alias address is assigned to the service adapter as an alias for the boot IP address. To set the IP alias proceed with the following section.

## Naming Scheme for IP Addresses

Using a standard naming scheme in your cluster environment will help you avoid confusion with the following:

- IP Addresses

Other IP addresses that may be set on the interface.

- Messages

Messages in the message browser originating from addresses other than the service address of the system.

Use the following naming scheme in your cluster environment:

<code>&lt;systemname&gt;_boot</code>	Boot address of a system
<code>&lt;systemname&gt;_svc</code>	Service address of a system
<code>&lt;systemname&gt;_stdby</code>	Standby address of a system
<code>&lt;systemname&gt;_mwa</code>	IP alias of a system

In this naming scheme, `<systemname>` is the name of the system as defined in the cluster configuration.

## To Set an IP Alias

Before running OV Performance Agent on a cluster node, you *must* set an IP alias on each system on which you wish to run the OV Performance Agent.

To set an IP alias, follow these steps:

- 1 Use the System Management Interface Tool (SMIT) menus.
- 2 In a shell, enter the following command:

```
smit tcpip
```

- 3 Select the following from the menu bar:

**Further Configuration -> Network Interface Selection -> Configure Aliases -> Add an IPV4 Network Alias**

- 4 Select the interface you want (for example, en0).
- 5 Enter values for the IP address and network mask.

## Configuring the OV Performance Agent Environment

The `<systemname>_boot`, `<systemname>_svc` and `<systemname>_stdby` IP addresses must be listed in the `/etc/default/ovpa` file. Add these IP addresses to the `RPC_UNSUPPORTED_NETADDRS` environment variable.

For example:

Node Name	IP address
node-svc	10.17.1.2
node-stdby	10.18.1.1
node-boot	10.17.1.1

```
RPC_UNSUPPORTED_NETADDRS=10.17.1.2:10.18.1.1:10.17.1.1
export RPC_UNSUPPORTED_NETADDRS
```

For information on how to modify environment variables in the `/etc/default/ovpa` file, refer to [The `/etc/default/ovpa` File](#) on page 29.

## Troubleshooting Hints

The following topics are described in this section:

- Possible problems

Problems you may encounter when running OV Performance Agent on a cluster node with workarounds.

- Fixing problems with IP aliases

Useful information on how to fix problems with IP aliases if you are using HACMP for AIX software.

### Possible Problems

After a cluster is switched on/off, OV Performance Agent servers cannot be restarted. The problem occurs in one of the following cases:

- If DCE was started prior to running the cluster when the cluster was still inoperative.

Workaround:

- a Stop your cluster software. For information on how to do that, refer to your HACMP documentation.
- b Stop the DCE daemon.
- c Run your cluster software. For information on how to do that, refer to your HACMP documentation.

- d Start the DCE daemon.
- e Make sure there is no OV Performance Agent server running on your system, else you will have to stop them.
- f Start the OV Performance Agent.
- If DCE was started prior to stopping the cluster when the cluster was still active.

Workaround:

- a Start your cluster software. For information on how to do that, refer to your HACMP documentation.
- b Stop the DCE daemon.
- c Stop your cluster software. For information on how to do this, refer to your HACMP cluster documentation.
- d Start the DCE daemon.
- e Make sure there is no OV Performance Agent server running on your system, else you will have to stop them.
- f Start OV Performance Agent.

### Fixing Problems with IP Aliases

Once you set the IP alias for OV Performance Agent on AIX, HACMP no longer works correctly. This problem applies to *all* events that deal with IP addresses (for example, acquire service address, acquire takeover address, swap adapter, and so on). This problem results from a flaw in the AIX operating system.

To fix AIX problems with IP aliases and HACMP, follow these steps:

- 1 Download and install the appropriate fixes for the AIX operating system.  
You can get the fixes with the IBM “FixDist” package or from the IBM website.



For systems where AIX operating system fixes have already been installed, refer to [To Reset Events on HACMP 4.2.2](#) on page 56.

To get the fixed versions of related packages, use the following APAR:

**IX78397**

- 2 Reset IP aliases on the network interface card.

Once you have installed the fixes to the AIX operating system, all HACMP events work, and the IP alias is set on the interface. However, the IP alias address no longer works after the `ifconfig` command is used to change the main IP address on the interface. For this reason, you have to reset the IP alias on the interface after each change of the IP address. For instructions, see [To Reset the IP Alias on the Network Interface Card](#).



You have to reset the IP alias on all cluster systems where OV Performance Agent is to be installed.

### [To Reset the IP Alias on the Network Interface Card](#)

To reset the IP alias on the interface where the service or boot IP address is set, use the following shell script (from here on referred to as `set_alias` script):

```
#!/bin/sh
# Specify MWA alias IP address below
ALIAS_IP="0.0.0.0"
SERVICE_IP=`/usr/sbin/cluster/utilities/cllsif -cSi \
    $LOCALNODENAME | grep ":service:.*:ether" | cut -d: -f7 |\
    uniq`
BOOT_IP=`/usr/sbin/cluster/utilities/cllsif -cSi \
    $LOCALNODENAME |\
    grep ":boot:.*:ether" | cut -d: -f7 | uniq`
INTERFACE=`/usr/sbin/cluster/utilities/clgetif -a
$SERVICE_IP`
if [ $? -ne 0 ]; then
    INTERFACE=`/usr/sbin/cluster/utilities/clgetif -a $BOOT_IP`
fi
if [ "$INTERFACE" != "" ]; then
    #IP has changed, set IP alias again on interface with
    SERVICE_IP
    /usr/sbin/ifconfig $INTERFACE $ALIAS_IP alias
fi
```

The `ALIAS_IP` variable should contain the same IP address you used to install the OV Performance Agent. If you copy the shell script to other systems in the cluster, make sure to change the `ALIAS_IP` variable. The shell script gets service and boot IP addresses for the local system, and sets the IP alias on the interface where either of the two was found.

In addition, you can use the `set_alias` script as the post-event script for the following HACMP events:

- Acquire service address
- Release service address
- Swap adapter

#### To Reset Events on HACMP 4.2.2

To reset events on HACMP 4.2.2, follow these steps:

- 1 Use the SMIT screens by entering the following command in a shell:  
**smit hacmp**
- 2 Select the following:  
**Cluster Configuration -> Cluster Resources -> Change/Show Cluster Events**
- 3 Select the appropriate option from the list, and fill in the Post-event Command field.

You can put the `set_alias` script in the following directory:

```
/usr/sbin/cluster/local
```

#### To Reset Events on HACMP 4.3.1

To reset events on HACMP 4.3.1, follow these steps:

- 1 Use the SMIT screens by entering the following command in a shell:  
**smit hacmp**
- 2 Go into the Cluster Events menu.
- 3 Select the following:  
**Cluster Configuration -> Cluster Resources -> Cluster Events**
- 4 Add the `set_alias` script to the Known Cluster Events list.
- 5 Select the following:  
**Define Custom Cluster Events -> Add a Custom Cluster Event**
- 6 Set the following:

```
Cluster Event Name Set to set_alias
```

```
Cluster Event Description Set to MWA set_alias
```



Cluster Event Script Filename Set to /usr/sbin/cluster/local/  
set\_alias

Then click **[OK]**.

- 7 Assign it to all appropriate events.

Press **Cancel** to go to the previous level. Then select **Change/Show Cluster Events**.

- 8 Select the appropriate option and enter **set\_alias** in the Post-event Command field for each event:
  - acquire service address
  - release service address
  - swap adapter

# 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 59.

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`):

```
/usr/lpp/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 `/usr/lpp/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 OVPA Documentation Set**

Document	File Name	Format
<i>HP OpenView Performance Agent for IBM RS/6000 systems Installation &amp; Configuration Guide</i>	ovpainst.pdf	/usr/lpp/perf/paperdocs/ovpa/C/
<i>HP OpenView Performance Agent for UNIX User's Guide</i>	ovpausers.pdf	/usr/lpp/perf/paperdocs/ovpa/C/
<i>HP OpenView Performance Agent for UNIX Data Source Integration Guide</i>	ovpadsi.pdf	/usr/lpp/perf/paperdocs/ovpa/C/
<i>HP OpenView Performance Agent &amp; GlancePlus for UNIX Tracking Your Transactions</i>	tyt.pdf	/usr/lpp/perf/paperdocs/ovpa/C/
<i>Application Response Measurement (ARM) API Guide</i>	arm2api.pdf	/usr/lpp/perf/paperdocs/arm/C/
<i>HP OpenView Performance Agent AIX Metric Definitions</i>	metaix.txt	/usr/lpp/perf/paperdocs/ovpa/C/
<i>HP OpenView Performance Agent Metrics list by Data Class for all operating systems</i>	mettable.txt	/usr/lpp/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 AIX, 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 metaix.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 communication 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

### CEC

Central Electronics Complex. A CEC is a single HMC-attached pSeries server, which can be divided into LPARs.

### 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

## A

- alarmdef file, 26, 61
- alarm generator, starting, 24
- alarms, 61
- authip file, 48
  - examples, 49
  - formatting, 49

## C

- client authentication, 48
- cluster configuration, 51
- coda.log file, 59
- communicating across a firewall, 39
- communication protocols, 10
- configuring
  - data sources, 58
  - OVPM and OVPA firewall communication, 40
  - OVPM C.03.00 and later with OVPA firewall communication, 40
  - OV Reporter and OVPA firewall communication, 44
- configuring data sources, 58

- configuring OVPA to run on a cluster node, 51
  - configuring OVPA environment, 52
  - naming IP addresses, 51
  - setting an IP alias, 52
  - troubleshooting hints, 53

## D

- data sources
  - configuring, 58
  - deleting, 58
  - DSI, 58
  - SCOPE, 58
  - scopeux, 58
- defining alarms, 61
- deleting data sources, 58
- disk space requirements, 10
- documentation
  - viewing on AIX, 63
  - viewing on the web, 63
- DSI data sources, 58

## E

- environment variables, 29, 30
- examples
  - README, 32
- extract program, 59

## F

### files

- /etc/default/mwa, 29
- alarmdef, 61
- ASCII, 63
- coda.log, 59
- parm, 60
- perflbd.rc, 26, 58
- status.scope, 32

### firewall

- communicating across, 33
- overview of OVPA communications
  - configuration, 34
- systems with multiple IP addresses, 39

### firewalls, 39

- communicating across, 39
- configuring OV Reporter and OVPA communication, 44
- configuring with OVPM C.03.00 and later, 40
- configuring with OV Reporter A.02.00 and later, 44

## H

### hardware requirements, 9

## I

### install.mwa script, 20

### installation procedures, 11

- installing from CD-ROM, 16
- installing with OV Operations installed on your system, 18

### installation requirements, 9

- disk space, 10
- hardware, 9

## M

### metric definitions, printing, 62

### mwa.remove script, 21

### mwa restart script, 26

### mwa script, 24, 27

### mwa scripts

- mwa restart, 28
- mwa restart alarms, 28
- mwa restart server, 27
- mwa stop, 27

## O

### OVPA

- stopping, 15

### ovpa scripts

- ovpa start, 25

### ovpa stop script, 15

### OV Performance Agent

- alarms, 61
- documentation set, 62
- how it works, 7
- removing, 21
- restarting, 27
- running on cluster node, 51
- starting, 24
- status files, 32
- stopping and restarting, 27

### OV Performance Manager

- configuring OVPM C.03.00 and later
  - with OVPA firewall communication, 40
- firewall configuration with OVPA, 40

### OV Reporter

- firewall configuration, 44

## P

### parm file, 60

- configuration, 60
- restarting, 26

- perflbd, 58
- perflbd.rc file, 26, 58
- perfstat command, 15
- printable files, 62
- printing documents, 62, 63

## R

- removing OV Performance Agent, 21
- Reporter
  - see OV Reporter, 44
- repository servers, 58
  - restarting, 26
- restarting
  - OV Performance Agent, 27
- restricting RPC addresses, 45
- RPC\_SUPPORTED\_NETADDRS
  - environment variable, 45

## S

- SCOPE default data source, 58
- scopeux
  - data sources, 58
  - starting, 24
- software requirements, 9
- starting
  - alarm generator, 24
  - OV Performance Agent, 24
  - scopeux, 24
  - using mwa, 26
  - using ovpa, 25
- status.scope file, 32

- status files
  - coda.txt, 32
  - status.alarmgen, 32
  - status.ls, 32
  - status.mi, 32
  - status.perfalarm, 32
  - status.perflbd, 32
  - status.rep\_server, 32
  - status.scope, 32
  - status.ttd, 32
- status files, OV Performance Agent, 32
- stopping
  - OV Performance Agent, 27
- stopping OVPA, 15
- stopping processes prior to installation, 15

## T

- target
  - nodes, 70

## V

- variables, 29, 30
  - MWA\_LLBD\_COMMAND, 30
  - MWA\_PROTOCOL, 30
  - MWA\_RPC\_INETADDR, 30
  - MWA\_START, 30
  - MWA\_START\_COMMAND, 30
  - RPC\_NOALIAS\_NETIFS, 31
  - RPC\_RESTRICTED\_PORTS, 31
  - RPC\_SUPPORTED\_NETADDRS, 31
  - RPC\_UNSUPPORTED\_NETADDRS, 31

