Legal Notices

Warranty.

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

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

Restricted Rights Legend.

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

Hewlett-Packard Company
United States of America

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

Copyright Notices.

©Copyright 2004 Hewlett-Packard Development Company, L.P.

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

Trademark Notices.

Adobe® is a trademark of Adobe Systems Incorporated.

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

UNIX® is a registered trademark of the Open Group.

Windows® and MS Windows® are U.S. registered trademarks of Microsoft Corporation.
### 1. OVO HTTPS Agent Overview

- **Introduction** ...................................................... 24
- **HP OpenView Operations HTTPS Agent Architecture** .......... 27
- **HTTPS Agent Platforms Supported with OVO 8.0** .......... 28
- **OVO Server Components and Processes** ........................ 29
- **New Processes on the OVO Management Server** ............ 29
- **Comparison of HTTPS and DCE Agents** ........................ 31
- **Configuration Deployment** ....................................... 31
- **Distribution Managers** .......................................... 32
- **Multiple Parallel Configuration Servers** ...................... 32
- **Comparison of Resource Requirements** .......................... 32
- **Comparison of Agent Performance** .............................. 33
- **Comparison of Agent Commands** ............................... 33
- **Comparison of Agent Processes** ................................ 34
- **Comparison of Troubleshooting Methods** ..................... 35
- **Generic Directory Structure on OVO Managed Nodes** ....... 36
- **HTTPS Communication Administration Commands in OVO** .... 37

### 2. Concepts of HTTPS Communication

- **HTTPS Communication in OVO** .................................. 42
- **Advantages** ...................................................... 43
- **Firewall Friendly** ............................................... 43
- **Secure** .......................................................... 44
- **Open** .......................................................... 45
- **Scalable** ........................................................ 45

### 3. Security Concepts

- **HTTPS-Based Security Components** .............................. 48
- **Certificates** ..................................................... 51
- **HP OpenView Certificate Server** ................................ 52
- **Certification Authority** ......................................... 52
- **Certificate Client** ............................................... 53
  - **Root Certificate Update and Deployment** ..................... 54
- **Environments Hosting Several Certificate Servers** .......... 55
- **Merging Two Existing MoM Environments** ..................... 56
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Certificate Handling for a Second OVO Management Server</td>
<td>59</td>
</tr>
<tr>
<td>Using a shared CA in MoM Environments</td>
<td>61</td>
</tr>
<tr>
<td>Remote Action Authorization</td>
<td>65</td>
</tr>
<tr>
<td>Server Configuration of Remote Action Authorization</td>
<td>65</td>
</tr>
<tr>
<td>Agents Running Under Alternative Users</td>
<td>70</td>
</tr>
<tr>
<td>Limitations of Running OVO Agents Under Alternative Users</td>
<td>71</td>
</tr>
<tr>
<td>Configuring an Agent to Run Under an Alternative User</td>
<td>72</td>
</tr>
<tr>
<td>Preparing the System Environment</td>
<td>72</td>
</tr>
<tr>
<td>Installing an Agent Using an Alternative User on UNIX Managed Nodes</td>
<td>73</td>
</tr>
<tr>
<td>Configuring the OVO Management Server For Agents Running Under Alternative Users</td>
<td>75</td>
</tr>
<tr>
<td>Changing the Default Port</td>
<td>76</td>
</tr>
<tr>
<td>Agent Profile</td>
<td>77</td>
</tr>
<tr>
<td>Upgrading and Patching an Agent Running Under an Alternative User</td>
<td>80</td>
</tr>
<tr>
<td>Copy To Node and Manually Install Later</td>
<td>80</td>
</tr>
<tr>
<td>Working with Sudo Programs on UNIX Agents</td>
<td>81</td>
</tr>
<tr>
<td>How to Setup a Sudo Program</td>
<td>82</td>
</tr>
<tr>
<td>A Comparison of DCE and HTTPS Alternative User Concepts</td>
<td>84</td>
</tr>
</tbody>
</table>

### 4. Concepts of Managing HTTPS Nodes

- Controlling HTTPS Nodes                                           | 88   |
- Configuration Deployment to HTTPS Nodes                            | 89   |
- Policy Management                                                  | 90   |
- Instrumentation Management                                          | 90   |
- Manual Installation of Policies and Instrumentation                 | 91   |
- HTTPS Agent Distribution Manager                                   | 92   |
- Configuration Push                                                  | 93   |
- Delta Distribution                                                  | 94   |
- Multiple Parallel Configuration Servers                             | 94   |
- Heartbeat Polling of HTTPS Nodes                                   | 96   |
- Reduce Network and CPU Load                                         | 96   |
- Remote Control of HTTPS Nodes                                      | 98   |

### 5. Working with HTTPS Nodes

- Configuring HTTPS Nodes                                            | 100  |
- Installing OVO Software Automatically on HTTPS Nodes               | 101  |
Configuring a Windows Installation Server ........................................ 109
Migrating a DCE Agent to an HTTPS Agent ...................................... 112
Upgrading in a MoM Environment ..................................................... 114
Migrating an HTTPS Agent to a DCE Agent .................................... 116
Installing Agents Manually ............................................................. 118
Certificate Installation Tips ......................................................... 118
To Install an Agent Manually from Package Files .......................... 119
Setting Variables in OVO ............................................................... 125
Installing Agents Using Clone Images .............................................. 127
Changing Hostnames and IP Addresses ........................................... 129
Manually Changing the Hostname or IP Address of a Managed Node 129
Automatically Changing the Hostname or IP Address of a Managed Node 134
Comparing Configured Nodes Against Name Resolution ................. 135
Proxies in OVO .............................................................................. 137
Configuring Proxies .................................................................. 139
Syntax ..................................................................................... 141
Manual Agent Installation Behind a HTTP Proxy ............................ 142
Setting Proxies on a Managed Node ................................................. 142
Setting Proxies on the OVO Management Server ......................... 143
De-installing Agents .................................................................. 144
De-installing Agents Automatically ............................................... 144
To De-install an Agent Manually .................................................. 144
De-installation Errors .................................................................. 144
Virtual Nodes in OVO ................................................................. 145
Terminology ............................................................................... 145
Virtual Node Concepts ................................................................. 146
Adding Virtual Nodes to OVO ......................................................... 146
Configuring Virtual Nodes using opcnodex1m) ............................... 147
Modifying Virtual Nodes in OVO .................................................... 148
Deleting Virtual Nodes from OVO ................................................. 148
Assigning Policies to Virtual Nodes in OVO ................................. 149
De-assigning Policies from Virtual Nodes in OVO ......................... 149
Deploying Policies to Virtual Nodes in OVO ................................. 150
Modifying Policy Configuration on Virtual Nodes in OVO ............. 150
Managing HTTPS Agents on DHCP Client Systems .................... 151
DHCP Settings in OVO ................................................................. 152
### A. Troubleshooting HTTPS-based Communication

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Troubleshooting</td>
<td>182</td>
</tr>
<tr>
<td>Troubleshooting Tools</td>
<td>182</td>
</tr>
<tr>
<td>Ping an HTTPS-Based Application</td>
<td>182</td>
</tr>
<tr>
<td>Display the Current Status of an HTTPS-Based Application</td>
<td>183</td>
</tr>
<tr>
<td>Display All Applications Registered to a Communication Broker</td>
<td>183</td>
</tr>
<tr>
<td>What String</td>
<td>184</td>
</tr>
<tr>
<td>List All Installed OV Filesets on an HTTPS Node</td>
<td>184</td>
</tr>
<tr>
<td>Standard TCP/IP Tools</td>
<td>186</td>
</tr>
<tr>
<td>RPC Calls Take Too Long</td>
<td>186</td>
</tr>
<tr>
<td>Logging</td>
<td>187</td>
</tr>
<tr>
<td>Communication Problems between Management Server and HTTPS Agents</td>
<td>188</td>
</tr>
<tr>
<td>Basic Network Troubleshooting</td>
<td>188</td>
</tr>
<tr>
<td>Basic HTTP Communication Troubleshooting</td>
<td>190</td>
</tr>
<tr>
<td>Troubleshooting Authentications and Certificates in HTTP Communication</td>
<td>197</td>
</tr>
<tr>
<td>Troubleshooting OVO Communication</td>
<td>202</td>
</tr>
<tr>
<td>Problems during Certificate Deployment</td>
<td>206</td>
</tr>
<tr>
<td>Invalid OvCoreIds on OVO Management Servers</td>
<td>207</td>
</tr>
<tr>
<td>Certificate Backup and Recovery in OVO</td>
<td>210</td>
</tr>
<tr>
<td>When to Backup Certificates</td>
<td>211</td>
</tr>
</tbody>
</table>

### B. Configuring HTTPS-based Communication
C. HTTPS Communication Architecture

Communication (Broker) Architecture ............................................... 226

D. Firewalls and HTTPS Communication

Firewall Scenarios ................................................................. 230
Contacting an Application on the Internet from an Intranet using an HTTP Proxy .... 230
Contacting an Application on the Internet from an Intranet without an HTTP Proxy .... 231
Contacting an Application within a Private Intranet from an OpenView Application on the Internet .......................................................... 231
Contacting an Application within a Private Intranet from an OpenView Application on the Internet without using HTTP Proxies ................................. 231

E. OVO 8.1 Quick Start Guide

OVO 8.1 Quick Start for OVO 7.x Users ........................................ 234
Printing History

The printing date and part number of the manual indicate the edition of the manual. The printing date will change when a new edition is printed. Minor changes may be made at reprint without changing the printing date. The part number of the manual will change when extensive changes are made.

Manual updates may be issued between editions to correct errors or document product changes. To ensure that you receive the updated or new editions, you should subscribe to the appropriate product support service. See your HP sales representative for details.

Table 1

<table>
<thead>
<tr>
<th>First Edition:</th>
<th>June 2004</th>
</tr>
</thead>
</table>
Conventions

The following typographical conventions are used in this manual.

Table 2  Typographical Conventions

<table>
<thead>
<tr>
<th>Font</th>
<th>Meaning</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Italic</strong></td>
<td>Book or manual titles, and man page names</td>
<td>Refer to the <em>OVO Administrator's Reference</em> and the <em>opc(1M)</em> manpage for more information.</td>
</tr>
<tr>
<td></td>
<td>Emphasis</td>
<td>You <em>must</em> follow these steps.</td>
</tr>
<tr>
<td></td>
<td>Variable that you must supply when entering a command</td>
<td>At the prompt, enter <code>rlogin username</code>.</td>
</tr>
<tr>
<td></td>
<td>Parameters to a function</td>
<td>The <em>oper_name</em> parameter returns an integer response.</td>
</tr>
<tr>
<td><strong>Bold</strong></td>
<td>New terms</td>
<td>The <strong>HTTPS agent</strong> observes...</td>
</tr>
<tr>
<td></td>
<td><strong>Computer</strong> Text and other items on the computer screen</td>
<td>The following system message appears:</td>
</tr>
<tr>
<td></td>
<td></td>
<td><em>Are you sure you want to remove current group?</em></td>
</tr>
<tr>
<td></td>
<td>Command names</td>
<td>Use the <em>grep</em> command ...</td>
</tr>
<tr>
<td></td>
<td>Function names</td>
<td>Use the <em>opc_connect()</em> function to connect ...</td>
</tr>
<tr>
<td></td>
<td>File and directory names</td>
<td><em>/opt/OV/bin/OpC/</em></td>
</tr>
<tr>
<td></td>
<td>Process names</td>
<td>Check to see if <em>opcmona</em> is running.</td>
</tr>
<tr>
<td></td>
<td>Window/dialog box names</td>
<td>In the <em>Add Logfile</em> window ...</td>
</tr>
<tr>
<td></td>
<td>Menu name followed by a colon (:). means that you select the menu, then the item. When the item is followed by an arrow (-&gt;), a cascading menu follows.</td>
<td>Select <strong>Actions</strong>: Filtering -&gt; All Active Messages from the menu bar.</td>
</tr>
</tbody>
</table>
### Table 2  
**Typographical Conventions (Continued)**

<table>
<thead>
<tr>
<th>Font</th>
<th>Meaning</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Computer Bold</strong></td>
<td>Text that you enter</td>
<td>At the prompt, enter <code>ls -l</code></td>
</tr>
<tr>
<td><strong>Keycap</strong></td>
<td>Keyboard keys</td>
<td>Press Return.</td>
</tr>
<tr>
<td><strong>[Button]</strong></td>
<td>Buttons in the user interface</td>
<td>Click [OK].</td>
</tr>
</tbody>
</table>
OVO Documentation Map

HP OpenView Operations (OVO) provides a set of manuals and online help that help you use the product and understand the concepts underlying the product. This section describes what information is available and where you can find it.

Electronic Versions of the Manuals

All manuals are available as Adobe Portable Document Format (PDF) files in the documentation directory on the OVO product CD-ROM.

With the exception of the OVO Software Release Notes, all manuals are also available in the following OVO web server directory:

http://<management_server>:3443/ITO_DOC/<lang>/manuals/*.pdf

In this URL, <management_server> is the fully qualified hostname of your management server, and <lang> stands for your system language, for example C for English and japanese for Japanese environments.

Alternatively, you can download the manuals from the following website:

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

Please watch this website regularly for the latest edition of the OVO Software Release Notes, which gets updated every 2-3 months with the latest news such as additionally supported operating system versions and latest patches.
OVO Manuals

This section provides an overview of the OVO manuals and their contents.

Table 3  OVO Manuals

<table>
<thead>
<tr>
<th>Manual</th>
<th>Description</th>
<th>Media</th>
</tr>
</thead>
<tbody>
<tr>
<td>OVO Installation Guide for the Management Server</td>
<td>Designed for administrators who install OVO software on the management server and perform initial configuration. This manual describes: • Software and hardware requirements • Software installation and de-installation instructions • Configuration defaults</td>
<td>Hardcopy, PDF</td>
</tr>
<tr>
<td>OVO Concepts Guide</td>
<td>Provides you with an understanding of OVO on two levels. As an operator, you learn about the basic structure of OVO. As an administrator, you gain insight into the setup and configuration of OVO in your own environment.</td>
<td>Hardcopy, PDF</td>
</tr>
<tr>
<td>OVO Administrator's Reference</td>
<td>Designed for administrator's who install OVO on the managed nodes and are responsible for OVO administration and troubleshooting. Contains conceptual and general information about the OVO DCE/NCS-based managed nodes.</td>
<td>PDF only</td>
</tr>
<tr>
<td>DCE Agent Concepts and Configuration Guide</td>
<td>Provides platform-specific information about each DCE/NCS-based managed node platform.</td>
<td>PDF only</td>
</tr>
<tr>
<td>HTTPS Agent Concepts and Configuration Guide</td>
<td>Provides platform-specific information about each HTTPS-based managed node platform.</td>
<td>PDF only</td>
</tr>
<tr>
<td>OVO Reporting and Database Schema</td>
<td>Provides a detailed description of the OVO database tables, as well as examples for generating reports from the OVO database.</td>
<td>PDF only</td>
</tr>
<tr>
<td>OVO Entity Relationship Diagrams</td>
<td>Provides you with an overview of the relationships between the tables and the OVO database.</td>
<td>PDF only</td>
</tr>
</tbody>
</table>
### Table 3  OVO Manuals (Continued)

<table>
<thead>
<tr>
<th>Manual</th>
<th>Description</th>
<th>Media</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>OVO Java GUI Operator's Guide</strong></td>
<td>Provides you with a detailed description of the OVO Java-based operator GUI and Service Navigator. This manual contains detailed information about general OVO and Service Navigator concepts and tasks for OVO operators, as well as reference and troubleshooting information.</td>
<td>PDF only</td>
</tr>
<tr>
<td><strong>Service Navigator Concepts and Configuration Guide</strong></td>
<td>Provides information for administrators who are responsible for installing, configuring, maintaining, and troubleshooting the HP OpenView Service Navigator. This manual also contains a high-level overview of the concepts behind service management.</td>
<td>Hardcopy PDF</td>
</tr>
<tr>
<td><strong>OVO Software Release Notes</strong></td>
<td>Describes new features and helps you:</td>
<td>PDF only</td>
</tr>
<tr>
<td></td>
<td>• Compare features of the current software with features of previous versions.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Determine system and software compatibility.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Solve known problems.</td>
<td></td>
</tr>
<tr>
<td><strong>OVO Supplementary Guide to MPE/iX Templates</strong></td>
<td>Describes the message source templates that are available for MPE/iX managed nodes. This guide is not available for OVO on Solaris.</td>
<td>PDF only</td>
</tr>
<tr>
<td><strong>Managing Your Network with HP OpenView Network Node Manager</strong></td>
<td>Designed for administrators and operators. This manual describes the basic functionality of HP OpenView Network Node Manager, which is an embedded part of OVO.</td>
<td>Hardcopy PDF</td>
</tr>
<tr>
<td><strong>OVO Database Tuning</strong></td>
<td>This ASCII file is located on OVO management server on the following location:</td>
<td>ASCII</td>
</tr>
<tr>
<td></td>
<td>/opt/OV/ReleaseNotes/opc_db.tuning</td>
<td></td>
</tr>
</tbody>
</table>
**Additional OVO-related Products**

This section provides an overview of the OVO-related manuals and their contents.

**Table 4  Additional OVO-related Manuals**

<table>
<thead>
<tr>
<th>Manual</th>
<th>Description</th>
<th>Media</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>HP OpenView Operations for UNIX Developer's Toolkit</strong></td>
<td>If you purchase the HP OpenView Operations for UNIX Developer's Toolkit, you receive the full OVO documentation set, as well as the following manuals:</td>
<td></td>
</tr>
<tr>
<td><em>OVO Application Integration Guide</em></td>
<td>Suggests several ways external applications can be integrated into OVO.</td>
<td>Hardcopy</td>
</tr>
<tr>
<td><em>OVO Developer's Reference</em></td>
<td>Provides an overview of all available application programming interfaces (APIs).</td>
<td>Hardcopy</td>
</tr>
<tr>
<td><strong>HP OpenView Event Correlation Designer for NNM and OVO</strong></td>
<td>If you purchase HP OpenView Event Correlation Designer for NNM and OVO, you receive the following additional documentation. Note that HP OpenView Event Correlation Composer is an integral part of NNM and OVO. OV Composer usage in the OVO context is described in the OS-SPI documentation.</td>
<td></td>
</tr>
<tr>
<td><em>HP OpenView ECS Configuring Circuits for NNM and OVO</em></td>
<td>Explains how to use the ECS Designer product in the NNM and OVO environments.</td>
<td>Hardcopy</td>
</tr>
</tbody>
</table>
# OVO Online Information

The following information is available online.

<table>
<thead>
<tr>
<th>Online Information</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HP OpenView Operations Administrator’s Guide to Online Information</td>
<td>Context-sensitive help system contains detailed help for each window of the OVO administrator Motif GUI, as well as step-by-step instructions for performing administrative tasks.</td>
</tr>
<tr>
<td>HP OpenView Operations Java GUI Online Information</td>
<td>HTML-based help system for the OVO Java-based operator GUI and Service Navigator. This help system contains detailed information about general OVO and Service Navigator concepts and tasks for OVO operators, as well as reference and troubleshooting information.</td>
</tr>
<tr>
<td>HP OpenView Operations Man Pages</td>
<td>Manual pages available online for OVO. These manual pages are also available in HTML format.</td>
</tr>
<tr>
<td></td>
<td>To access these pages, go to the following location (URL) with your web browser:</td>
</tr>
<tr>
<td></td>
<td>http://&lt;management_server&gt;:3443/ITO_MAN</td>
</tr>
<tr>
<td></td>
<td>In this URL, the variable <code>&lt;management_server&gt;</code> is the fully qualified hostname of your management server. Note that the appropriate man pages for the OVO HTTPS-agent are installed on each managed node.</td>
</tr>
</tbody>
</table>
About OVO Online Help

This preface describes online documentation for the HP OpenView Operations (OVO) Motif and Java operator graphical user interfaces (GUIs).

Online Help for the Motif GUI

Online information for HP OpenView Operations (OVO) Motif graphical user interface (GUI) consists of two separate volumes, one for operators and one for administrators. In the operator’s volume, you will find the HP OpenView OVO Quick Start describing the main operator windows.

Types of Online Help

The operator and administrator volumes include the following types of online help:

- **Task Information**
  Information you need to perform tasks, whether you are an operator or an administrator.

- **Icon Information**
  Popup menus and reference information about OVO icons. You access this information with a right-click of your mouse button.

- **Error Information**
  Information about errors displayed in the OVO Error Information window. You can access context-sensitive help when an error occurs. Or you can use the number provided in an error message to perform a keyword search within the help system.

- **Search Utility**
  Index search utility that takes you directly to topics by name.

- **Glossary**
  Glossary of OVO terminology.

- **Help Instructions**
  Instructions about the online help system itself for new users.
Printing Facility
Printing facility, which enables you to print any or all topics in the help system. (An HP LaserJet printer or a compatible printer device is required to print graphics.)

To Access Online Help
You can access the help system in any of the following ways:

- **F1 Key**
  Press F1 while the cursor is in any active text field or on any active button.

- **Help Button**
  Click [Help] in the bottom of any window.

- **Help Menu**
  Open the drop-down Help menu from the menu bar.

- **Right Mouse Click**
  Click a symbol, then right-click the mouse button to access the Help menu.

You can then select task lists, which are arranged by activity, or window and field lists. You can access any topic in the help volume from every help screen. Hyperlinks provide related information on other help topics.

You can also access context-sensitive help in the Message Browser and Message Source Templates window. After selecting Help: On Context from the menu, the cursor changes into a question mark, which you can then position over the area about which you want help. When you click the mouse button, the corresponding help page is displayed in its help window.

Online Help for the Java GUI and Service Navigator
The online help for the HP OpenView Operations (OVO) Java graphical user interface (GUI), including Service Navigator, helps operators to become familiar with and use the OVO product.
Types of Online Help

The online help for the OVO Java GUI includes the following information:

- **Tasks**
  Step-by-step instructions.

- **Concepts**
  Introduction to the key concepts and features.

- **References**
  Detailed information about the product.

- **Troubleshooting**
  Solutions to common problems you may encounter while using the product.

- **Index**
  Alphabetized list of topics to help you find the information you need quickly and easily.

To View a Topic

To view any topic, open a folder in the left frame of the online documentation window, then click the topic title. Hyperlinks provide access to related help topics.

To Access Online Help

To access the help system, select Help: Contents from the menu bar of the Java GUI. A web browser opens and displays the help contents.

**NOTE**

To access online help for the Java GUI, you must first configure OVO to use your preferred browser.
1 OVO HTTPS Agent Overview
Introduction

From OVO 8.0, the new HTTPS agent software is available for highly secure communication between OVO management servers and their managed nodes. HTTPS agents are generally used and administered in the same way as DCE-based agents. Applications are launched in the same way. Command line interfaces, such as `opcragt`, can be used for all managed nodes. All functionality that is available with DCE-based agents is also available with HTTPS agents unless explicitly stated otherwise.

Policies for HTTPS agents are created, assigned and deployed in a similar way as templates for DCE-based agents. For example, heartbeat polling of nodes results in the same type of status messages and are displayed in a very similar way in the message browser. Figure 1-1 illustrates a typical environment managed by HP OpenView Operations.

However, the HTTPS agents have many advantages and benefits over DCE-based agents. These are described in the following chapters.

Figure 1-1: A Typical OVO Managed Environment
HTTPS-based communication provides you with the following major advantages:

- No more need for DCE-RPC technology on the most commonly used managed node operating system platforms.
- Simple management through firewalls with configurable, single-port, secure communication using, open, HTTPS-based communication techniques. Restrict outside access to dedicated HTTP proxies and reduce port usage by multiplexing over HTTP proxies.
- Out-of-the-box Internet Secure Communication using SSL/PKI encryption with server and client certificates for authentication.
- Communication is based on standard Web technologies (HTTP, SOAP, Proxies, SSL, …), available in every environment today, and familiar to every IT administrator.
- No need for additional investments (training, additional software such as DCE).

Additional advantages available with OVO 8.0 include:

- The OVO management server can simultaneously manage HTTPS and OVO 7.x DCE managed node systems.
- New OVO message format based on XML and SOAP used for message security from the HTTPS agent to the OVO Server.
- IP independence/dynamic IP (DHCP). Managed nodes can be identified by their unique OvCoreID and not necessarily by their IP addresses.
- Duplicate IP support will be available for both HTTPS and DCE agents.
- New OpenView consistent control and deployment mechanism.
- New OpenView consistent logging capability.
- New OpenView consistent tracing capability.
Figure 1-2 illustrates an example of the different communication types in OVO.

Figure 1-2 Communication Overview in HP OpenView Operations

1. Socket communication is used to communicate with the OVO Java GUI. If OVAS is installed, Socket communication with SSL is used.
HP OpenView Operations HTTPS Agent Architecture

The following graphics illustrate the architecture of the HTTPS communication in OVO.

Figure 1-3 HTTPS Agent Components and Responsibilities
HTTPS Agent Platforms Supported with OVO 8.0

- HP-UX 11.0, 11.11, 11.23
- Solaris 7, 8, 9
- RedHat EL 2.1, EL 3.0, 8.0, 9.0
- SuSE 8.0, 8.1, 8.2
- Debian: 3.0
- Turbolinux: 8.0
- Tru64 5.1A, 5.1B
- AIX 5.1, 5.2

---

1. For the most current list of supported managed node platforms, refer to the latest version of the OVO Release Notes. This document is available in pdf format from: http://ovweb.external.hp.com/lpe/doc_serv/ under operations for UNIX, version 8.x. Select the operating system of your management server and all the related documentation will be listed.
OVO Server Components and Processes

The following server components communicate as RPC clients with the HTTPS agent:

- **ovoareqsdr** to send action request and to do heartbeat polling.
- **opcragt** to perform remote control and to do primary manager switches.
- **opcbbcdist** controls the configuration deployment to HTTPS nodes. The deployer is used during remote agent installation.

HTTPS-based communication RPC servers are:

- **ovbbccb** (communication broker).
- **opcmsgrb** (message receiver for HTTPS agents).
- **ovcs** (the security certificate server).

New Processes on the OVO Management Server

A number of new processes are introduced on the OVO management server. The command `opcsv -status` lists all processes that are relevant to OVO, but excludes Oracle and NNM processes. The call displays the following new processes:

- **opcbbcdist**: configuration deployment to HTTPS nodes. Analogous to `opcdistm` for DCE nodes. Both processes are controlled by `opctlm`.
- **opcmsgrb**: message receiver for HTTPS nodes. Analogous to `opcmsgrd` for DCE nodes. Both processes controlled by `ovoareqsdr`.
- **ovcd**: control daemon; self-controlled.
- **ovbbccb**: communication broker; controlled by `ovcd`.
- **ovdepl**: configuration and deployment process; controlled by `ovcd`.
- **ovcs**: server extension to handle certificate requests; controlled by `ovcd`.
- **opccsad**: OVO certificate server adapter; controlled by `opctlm`.
- **TraceServer**: OVO trace server.
When calling `ovstop ovoacomm`, no core OpenView processes are stopped. This includes also the ovcs server extensions. To stop all core OpenView processes, you must enter the command:

`ovstop ovctrl`

To terminate all core OpenView processes, enter the command:

`ovc -kill`

This also stops the OVO agent on the management server node.
Comparison of HTTPS and DCE Agents

Configuration Deployment

Configuration deployment to HTTPS agents differs slightly from that of DCE-based nodes:

- Policies are used by HTTPS agents. These refine and replace the Templates used by DCE-based agents.

  Policies are pushed out by the OVO management server. Templates for DCE agents are pulled by the OVO distribution agent. When the OVO management server system is located inside the trusted environment, policy deployment to managed nodes across a firewall is outbound only.

- Instrumentation is the single term used by HTTPS agents for Actions, Commands, and Monitors. All scripts and binaries are stored in a common instrumentation directory.

- A configuration parameter schema with a name-value pair policy type for HTTPS agents replaces nodeinfo and opcinfo files.

- mgrconf file is enhanced for HTTPS agents by a role model-based security authorization mechanism allowing the deployment of policies and instrumentation from more than one OVO management server.

For detailed information about HTTPS agent configuration management, refer to “Configuration Deployment to HTTPS Nodes” on page 89.
Distribution Managers

`opcbbcdist` is the configuration management adapter between the OVO management server and the HTTPS agents. Its main functions are:

- Convert existing templates into policies.
- Convert ECS templates and the associated circuits into policies.
- Convert node properties into the format used on HTTPS nodes. This replaces the `nodeinfo` file found on DCE nodes.

`opcbbcdist` only accepts requests from the OVO management server. `opcdistm`, the configuration management adapter between the OVO management server and the DCE agents accepts requests from the distribution agent (`opcdista`) of the DCE managed nodes.

Multiple Parallel Configuration Servers

Multiple parallel configuration servers are supported for HTTPS nodes through an owner concept for policies.

Comparison of Resource Requirements

Table 1-1 OVO Agent Footprint

<table>
<thead>
<tr>
<th>Description</th>
<th>HTTPS Agent</th>
<th>DCE Agent</th>
</tr>
</thead>
<tbody>
<tr>
<td>RAM</td>
<td>☺</td>
<td>☺</td>
</tr>
<tr>
<td>CPU</td>
<td>☺</td>
<td>☻</td>
</tr>
<tr>
<td>Disk</td>
<td>☻</td>
<td>☻</td>
</tr>
</tbody>
</table>

NOTE

The managed node footprint for the HTTPS agent becomes progressively more favorable as the number of new OpenView products installed increases. These products share the underlying OV infrastructure and so significantly less software needs to be installed and run as compared to traditionally designed software.
### Comparison of Agent Performance

<table>
<thead>
<tr>
<th>Description</th>
<th>HTTPS Agent</th>
<th>DCE Agent</th>
</tr>
</thead>
<tbody>
<tr>
<td>OVO agent binary installation</td>
<td>Full - 😊</td>
<td>Full - 😊</td>
</tr>
<tr>
<td></td>
<td>Patch - 😊</td>
<td>Patch - 😊</td>
</tr>
<tr>
<td>Policy and instrumentation deployment</td>
<td>Full - 😊</td>
<td>Full - 😊</td>
</tr>
<tr>
<td></td>
<td>Delta - 😊</td>
<td>Delta - 😊</td>
</tr>
<tr>
<td>OVO message throughput</td>
<td>😊</td>
<td>😊</td>
</tr>
</tbody>
</table>

### Comparison of Agent Commands

<table>
<thead>
<tr>
<th>Description</th>
<th>HTTPS Agent</th>
<th>DCE Agent</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start, stop, status, and control of the OVO agent</td>
<td>ovc</td>
<td>opcagt</td>
</tr>
<tr>
<td></td>
<td>opcagt wrapper</td>
<td></td>
</tr>
<tr>
<td>Policy/template management</td>
<td>ovpolicy</td>
<td>opctemplate</td>
</tr>
<tr>
<td></td>
<td>opctemplate wrapper</td>
<td></td>
</tr>
<tr>
<td>Local configuration settings</td>
<td>ovconfget</td>
<td>nodeinfo file</td>
</tr>
<tr>
<td></td>
<td>ovconfchg</td>
<td>opcinfo file</td>
</tr>
<tr>
<td></td>
<td>Configuration parameter schema with a name-value pair policy type.</td>
<td>Configuration files</td>
</tr>
<tr>
<td></td>
<td>ovconfget/set</td>
<td></td>
</tr>
<tr>
<td>Remote agent control from OVO server</td>
<td>opcragt</td>
<td>opcragt</td>
</tr>
<tr>
<td></td>
<td>ovconfget/set</td>
<td></td>
</tr>
</tbody>
</table>
### Comparison of Agent Processes

Table 1-4  OVO Agent Process Comparison

<table>
<thead>
<tr>
<th>Description</th>
<th>HTTPS Agent</th>
<th>DCE Agent</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start, stop, and control of the OVO agent</td>
<td>ovcd</td>
<td>opcctla</td>
</tr>
<tr>
<td>Policy and instrumentation deployment</td>
<td>ovconfd</td>
<td>opcdista</td>
</tr>
<tr>
<td>Communication</td>
<td>ovbbcccb</td>
<td>llbserver</td>
</tr>
<tr>
<td></td>
<td>HTTPS-RPC server using one configurable port. Default: 383.</td>
<td>dced, rpcd, or llbd on fixed port 135.</td>
</tr>
<tr>
<td></td>
<td>opcctsa - Certificate Adapter</td>
<td></td>
</tr>
<tr>
<td></td>
<td>ovccd - Certificate client</td>
<td></td>
</tr>
<tr>
<td>HTTPS agent configuration adapter</td>
<td>opchbbcdist</td>
<td>n.a.</td>
</tr>
<tr>
<td>Message agent</td>
<td>opcmsga</td>
<td>opcmsga</td>
</tr>
<tr>
<td>Monitor agent</td>
<td>opcmona</td>
<td>opcmona</td>
</tr>
<tr>
<td>Embedded Performance Component</td>
<td>coda</td>
<td>coda</td>
</tr>
<tr>
<td>Logfile encapsulator</td>
<td>opcle</td>
<td>opcle</td>
</tr>
<tr>
<td>Message Interceptor</td>
<td>opcmsgi</td>
<td>opcmsgi</td>
</tr>
<tr>
<td>SNMP Trap Interceptor</td>
<td>opctrapi</td>
<td>opctrapi</td>
</tr>
<tr>
<td>Event Correlation</td>
<td>opceca</td>
<td>opceca</td>
</tr>
<tr>
<td>ECS Annotate Server</td>
<td>opcecaas</td>
<td>opcecaas</td>
</tr>
</tbody>
</table>
Comparison of Troubleshooting Methods

Table 1-5  OVO Agent Troubleshooting Comparison

<table>
<thead>
<tr>
<th>Description</th>
<th>HTTPS Agent</th>
<th>DCE Agent</th>
</tr>
</thead>
</table>
| Tracing     | ovtrcadm\(^a\)  
              trcmon  
              ovtrcadm  
              ovtrccfg  
              TraceServer  
              Tracing is more powerful but there is some increased complexity associated with the greater functionality. | opcagt -trace |

\(^a\) Tracing capabilities of the HTTPS agent are described in detail in the dedicated document *HP OpenView Operations - Tracing Concepts and User's Guide.*
Generic Directory Structure on OVO Managed Nodes

The files associated with the HTTPS agent are found in four directory structures:

- **<OVInstallDir>**
  - HP-UX, Solaris, Linux: /opt/OV
  - Tru64: /usr/opt/OV
  - AIX: /usr/lpp/OV
  - Windows: <ProgramFilesDir>\HP OpenView

  This directory contains static files that are installed from the product media and never change, for example, executables. Since these files never change, you can mount <OVInstallDir> as “read-only” for increased security in highly sensitive environments. It is not necessary to back up these files as they can be re-installed from the product media.

  All other files change during operation and must be backed up regularly.

- **<OVDataDir>**
  - HP-UX, Solaris, Linux, AIX: /var/opt/OV
  - Tru64: /usr/var/opt/OV
  - Windows: <ProgramFilesDir>\HP OpenView\data

  This directory contains data files that are used only on the local system.
HTTPS Communication Administration Commands in OVO

HTTPS Communication can be controlled using the following commands.

On the OVO Management Server and Managed Nodes:

- **ovcoreid** (OpenView Unique System Identifier)
  
The `ovcoreid` command is used to display existing OvCoreId value and, in addition, create and set new OvCoreId values on the local node.
  
  For details of how to use this tool, refer to the `ovcoreid(1)` man page.

- **ovc** (OpenView Process Control)
  
  `ovc` controls starting and stopping, event notification, and status reporting of all components registered with the OpenView Control service, `ovcd`. A component can be a server process, an agent (for example, the Performance Agent or the Discovery Agent), an event interceptor, or an application delivered by an integrator.
  
  For details of how to use this tool, refer to the `ovc(1)` man page.

- **bbcutil**
  
  The `bbcutil` command is used to control the OV Communication Broker.
  
  For syntax information and details of how to use this tool, refer to the `bbcutil(1)` man page.

- **ovconfget**
  
  Installed OpenView components have associated configuration settings files that contain one or more namespaces and apply system wide or for a specified High Availability Resource Group. A namespace is a group of configuration settings that belong to a component. All configurations specified in the settings files are duplicated in the settings.dat configuration database.
For each specified namespace, `ovconfget` returns the specified attribute or attributes and writes them to `stdout`. Used without arguments, `ovconfget` writes all attributes in all namespaces to `stdout`.

For details of how to use this tool, refer to the `ovconfget(1)` man page.

- **ovconfchg**

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

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

  For details of how to use this tool, refer to the `ovconfchg(1)` man page.

- **ovpolicy**

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

  You use `ovpolicy` to install, remove, enable, and disable local policies and templates. For details of how to use this tool, refer to the `ovpolicy(1)` man page.
On Managed Nodes:

- **ovcert**
  
  The `ovcert` command is used to manage certificates on an HTTPS node through the Certificate Client. You can execute tasks such as initiating a new certificate request to the Certificate Server, adding node certificates and importing the private keys, adding certificates to the trusted root certificates, and checking the certificate status.

  For details of how to use this tool, refer to the `ovcert(1)` man page.

On the OVO Management Server:

- **opccsacm** (Certificate Server Adapter Control Manager)
  
  The `opccsacm` command is used to issue new node certificates and installation keys manually on the HP OpenView server. It also modifies the OVO database to reflect the changes made by certificate management actions.

  For details of how to use this tool, refer to the `opccsacm(1m)` man page.

- **opccsa** (Certificate Server Adapter)
  
  The `opccsa` command is used to list the pending certificate requests, map certificate requests to target nodes from the OVO database, grant, deny and delete specified certificate requests.

  For details of how to use this tool, refer to the `opccsa(1m)` man page.
Chapter 2

Concepts of HTTPS Communication
 HTTPS Communication in OVO

HTTPS 1.1 based communications is the latest communication technology used by HP OpenView products and allows applications to exchange data between heterogeneous systems.

OpenView products using HTTPS communication can easily communicate with each other, as well as with other industry-standard products. It is also now easier to create new products that can communicate with existing products on your network and easily integrate with your firewalls and HTTP-proxies. Figure 2-1 illustrates an example of HTTPS communication.

Figure 2-1 Communication Overview in HP OpenView Operations

---

1. Socket communication is used to communicate with the OVO Java GUI. If OVAS is installed, Socket communication with SSL is used.
Advantages

HTTPS communication provides the following major advantages:

- Firewall Friendly
- Secure
- Open
- Scalable

Firewall Friendly

More and more organizations need to cross firewalls in a safe, secure, and easily manageable way. Most of these organizations are very familiar and comfortable with HTTP, HTTP proxies, and firewalls. Their IT environments are already configured to allow communication through HTTP proxies and firewalls. By focusing on technology that is already a part of most IT infrastructures, it helps you to be more efficient and effective, without the need for new training. The end result reduces support and maintenance costs, while simultaneously creating a highly secure environment without significant effort.

Figure 2-2 illustrates crossing a firewall using HTTPS-communication.
Secure

HP OpenView’s HTTPS communication is based on the TCP/IP protocol, the industry standard for reliable networking. Using the Secure Socket Layer (SSL) protocol, HTTPS communication uses authentication to validate who can access data, and encryption to secure data exchange. Now that businesses are sending and receiving more transactions across the Internet and private intranets than ever before, security and authentication assume an especially important role.

HP OpenView’s HTTPS communication meets this goal through established industry standards. HTTP protocol and SSL encryption and authentication insure data integrity and privacy. By default, data is compressed, ensuring that data is not transmitted in clear text format, even for non-SSL connections.

In addition:

- All remote messages and requests arrive through the Communication Broker, providing a single port entry to the node.
- Restricted bind port range can be used when configuring firewalls.
- Configure one or more standard HTTP proxies to cross a firewall or reach a remote system when sending messages, files or objects.

Figure 2-3 illustrates crossing firewalls using standard HTTP proxies.

Figure 2-3  Crossing a Firewall using External HTTPS Proxies
To work with HTTPS communication and proxies, you will need to:

- Configure HTTP proxy servers.
- Implement SSL encryption.
- Establish server side authentication with server certificates.
- Establish client side authentication with client certificates.

How you do this in HP OpenView is described in the following sections.

**Open**

HP OpenView’s HTTPS communication is built on the industry standard HTTP 1.1 protocol and SSL sockets. HP OpenView’s adherence to open standards, such as HTTP, SSL and SOAP, allows you to maximize the use of your current HTTP infrastructure. For example, content filtering (without SSL and compression) using HTTP messages allows you to securely configure firewalls. Security is best implemented in layers and not just in one single location. Content filtering is a powerful tool used to add that extra layer of security.

HTTP proxies are widely used in today’s networks. They are workhorses to help safely bridge private networks to the Internet. The use of HTTP allows HP OpenView to slot into and take advantage of current infrastructures.

**Scalable**

HP OpenView’s HTTPS communication is designed to perform well, independent of the size of the environment and the number of messages sent and received. HP OpenView’s HTTPS communication can be configured to suit the environment within which it is to work. Large applications are able to handle many simultaneous connections while consuming the minimum of resources. If the maximum number of configured connections is exceeded, an entry in a logfile is created from which a warning message can also be raised.
Concepts of HTTPS Communication

Advantages
3 Security Concepts
HTTPS-Based Security Components

Managed nodes must have a valid, industry standard, X509 certificate issued by the HP OpenView Certificate Server to be able to communicate with HP OpenView management servers. Certificates, signed by 1024 bit keys, are required to identify nodes in a managed environment using the Secure Socket Layer (SSL) protocol. The “SSL handshake” between two nodes only succeeds if the issuing authority of the certificate presented by the incoming node is a trusted authority of the receiving node. The main communication security components responsible for creating and managing certificates are:

- HP OpenView Certificate Server
- HP OpenView Key Store
- HP OpenView Certificate Client

Figure 3-1 illustrates these components:

Figure 3-1  Components of Authenticated Communication
Each system hosting an HTTPS agent is allocated a unique identifier value for the parameter, `OvCoreId`, created during installation of the HP OpenView software on that system.

**NOTE**
After the `OvCoreId` for an HTTPS managed node has been created, it does not change, even if the hostname or the IP address, for example through DHCP, of the system is changed.

For each OpenView system (managed node or server) `OvCoreId` is used as a unique identifier and is contained in the corresponding node certificate. `OvCoreId` is allocated its value during installation.

Figure 3-2 illustrates an environment for authenticated communication:

**Figure 3-2**  Environment for Authenticated Communication
1. A server system hosts the Certificate Server, which contains the needed certification authority (CA) functionality.

2. Every system has a certificate that was signed by the Certificate Server with the certification authority private key.

3. The server system also needs a certificate to prove its identity.

4. Every system has a list of trusted root certificates, which must contain at least one certificate. The trusted root (CA) certificates are used to verify the identity of the communication partners; a communication partner is only trusted if the presented certificate can be validated using the list of trusted certificates.

A list of trusted root certificates is required, when the certificate client is being managed by more than one HP OpenView management server. For instance, when an OVO HTTPS managed node is managed simultaneously by multiple OVO management servers.
Certificates

There are two types of certificates:

- Root certificates
- Node certificates

A root certificate is a self-signed certificate, containing the identity of the certification authority of the certificate server. The private key belonging to the root certificate is stored on the certificate server system and protected from unauthorized access. The certification authority uses its root certificate to digitally sign all certificates.

Every HTTPS node in the managed environment receives a node certificate issued by a certificate server, a corresponding private key stored in the file system and the root certificates valid in its environment. The certificate client running on the node ensures this.

NOTE

A node certificate contains the unique identity OvCoreId. The following is an example of an OvCoreId:

d498f286-aa97-4a31-b5c3-806e384fcf6e

Each node can be securely authenticated through its node certificate. The node certificate can be verified by all other nodes in the environment using the root certificate(s) to verify the signature.

Node certificates are used to establish SSL-based connections between two HTTPS nodes that use client and server authentication, and can be configured to encrypt all communication.

The ovcert tool provided by the certificate client can be used to list the contents of the Key Store or to show information about an installed certificate. The ovcert tool is described in the ovcert man page.
HP OpenView Certificate Server

The certificate server is responsible for the following:

- Creating and installing self-signed root certificates.
- Importing self-signed root certificates from the file system.
- Storing the private keys of root certificates.
- Granting or denying certification requests.
- Creating a new certificate and a corresponding private key or creating an installation key for manual certificate installation.
- Offering a service for clients to automatically retrieve trusted root certificates.

Certification Authority

NOTE

Every OVO management server is automatically configured as a Certificate Authority. The default setting for `sec.cm.client:CERTIFICATE_SERVER` for every agent is its own OVO management server.

The certification authority is part of the certificate server and is the center of trust in certificate management. Certificates signed by this certification authority will be regarded as valid certificates and therefore be trustworthy. The certification authority must be hosted in a highly secure location. By default, it is installed on the system hosting the HP OpenView management server, for example the OVO management server system.

Since the certification authority is the root of trust, it operates with a self-signed root certificate. This root certificate and the corresponding private key are created and stored on the file system with the level of protection to allow the certification authority to operate. After the certification authority is successfully initialized, it is responsible for signing granted certificate requests using its root certificate.
Certificate Client

The certificate client runs on a managed node and acts as the counterpart of the certificate server’s certificate request handler.

The certificate client operates as follows:

- The certificate client checks whether the node has a valid certificate.
- If the node has no certificate, the certificate client generates a new public and private key pair and creates a certificate request based on the unique identity (OvCoreId value) of the node. This certificate request is sent to the certificate server together with any additional node properties and the certificate client waits for a response.

The additional node properties, for example DNS name and IP address of the node are intended to be used as additional information that, on the certificate server, should help to determine from which system in the environment a certificate request comes and to decide whether this request should be granted.

- After receiving the new certificate, it is installed on the node. After being installed, the certificate client can ensure that all HTTPS-based communication uses this certificate.

If the request is not successfully processed, a descriptive error is logged and the associated status is set.

In addition, the certificate client does the following:

- It can be triggered to contact a certificate server to update its trusted root certificates, for example, using the command line tool ovcert. Refer to the ovcert man page for details.

- It supports the import of a node certificate and the corresponding private key from the file system with its command line interface ovcert. For more details see “Certificate Generation for Manual Certificate Deployment” on page 163 and “Manual Certificate Deployment with Installation Key” on page 168. Manual certificate installation is used to improve security on sensitive systems.

- It supports the import of trusted root certificates.

- It provides status information. Status includes OK, valid certificate, no certificate, certificate requested, and certificate request denied.
Root Certificate Update and Deployment

It may be necessary to update the trusted root certificates of one or more nodes, for example, in environments hosting several HP OpenView certificate servers.

It is possible to supply all currently trusted root certificates to certificate clients in a secured way. It is usually sufficient to supply the root certificate of the certification authority. However, it may be necessary to deploy one or more additional root certificates to selected certificate clients, for example when there is more than one certification authority in the environment.

The certificate client allows triggering the “trusted root certificates update” through the command line tool ovcert. Refer to the ovcert man page.
Environments Hosting Several Certificate Servers

It is possible that a managed environment has more than one certificate server. This situation would arise if two existing managed environments, both having an operating certificate server are joined to form a single environment. This is termed merge.

Both certificate servers are each using a self-signed root certificate. As a result, all clients belonging to one certificate server do not trust any client belonging to the other. This is solved by adding the root certificate of each certificate server to the trusted root certificates of the other certificate server. Finally, all clients in the managed environment are triggered to receive the updated root certificate list from their certificate server.

NOTE

In a merge, it is also an option to only create a one-way trust. This may be very useful in a scenario where a number of groups of nodes are managed by their own trusted management servers. However, one of these management servers may be used as an escalation server and be able to manage any node in any of the sub groups. However, the other management servers are not trusted by the nodes of the escalation server and so they can only be managed by the escalation server.

If an agent is managed by multiple management servers some certificate management configuration must be made. By default, every OVO server has its own Certificate Authority and the agent trusts only certificates subscribed by this authority. For MoM environments, you must establish a trust between two or more managers so that their environments are able to communicate with each other.

The common scenarios are:

- “Merging Two Existing MoM Environments”
- “Certificate Handling for a Second OVO Management Server”
- “Using a shared CA in MoM Environments”

These scenarios are discussed in greater detail in the following sections.
Merging Two Existing MoM Environments

Assume you have an environment belonging to server M1 with the agents AM1 and the second of M2 with AM2. Assume that each server has its own Certificate Authority.

Complete the following steps to merge the environments:

**NOTE**

HA environments and non-HA environments are handled in the same way. The following steps are valid for both types of installations.

1. Synchronize the trusted certificates on the management servers: M1 gets the root certificates of M2 and M2 the root certificate of M1.
   a. On OVO management server M1, enter the command:
      
      `ovcert -exporttrusted -ovrg server -file <my_file>`

   b. Copy `<my_file>` to the management server M2, for example using ftp.

   c. Enter the following command on M2:
      
      `ovcert -importtrusted -ovrg server -file <my_file>`

   d. Repeat the procedure for management server M2.

   e. To verify that M1 and M2 have the root certificate of the other, on both management server systems, execute the command:
      
      `ovcert -list`

      Two trusted certificates should be listed.

2. Configure other management server as regular nodes in the OVO node bank. M1 must be added to the node bank of M2 with its coreid and M2 must be added to the node bank of M1 with its coreid.

   a. Add node M1 in the node bank of M2 and M2 in the node bank of M1 as follows:
      
      In the Administrator's GUI, select:
      
      Action -> Node -> Add

      Note: You can also use the command line tool:
      
      On node M1, enter the command:
Security Concepts

Environments Hosting Several Certificate Servers

```bash
opcnode -add_node node_name=<M2> \net_type=<network_type> mach_type=<machine_type> \group_name=<node_group_name>
```

On M2, enter the command:
```
opcnode -add_node node_name=<M2> \net_type=<network_type> mach_type=<machine_type> \group_name=<node_group_name>
```

b. M1's coreid must be stored in M2's database:

On M1, call the `ovcoreid` command to display the coreid of M1:
```
ovcoreid
```

Note down the displayed value.

On M2, call the `opcnode` command to add M1's coreid into M2's database:
```
opcnode -chg_id node_name=<M1> id=<core_id_of_M1>
```

c. M2's coreid must be stored in M1's database:

On M2, call the `ovcoreid` command to get coreid of M2:
```
ovcoreid
```

Note down the displayed value.

On M1, call the `opcnode` command to add M2's coreid into M1's database:
```
opcnode -chg_id node_name=<M2> id=<core_id_of_M2>
```

You can verify that the nodes have been correctly added to the databases by executing the following commands:

a. On M1, enter the command:
```
opcnode -list_id node_list=<M2>
```

The coreid of node M2 should be displayed.

b. On M2, enter the command:
```
opcnode -list_id node_list=<M1>
```

The coreid of node M1 should be displayed.
NOTE

Do not forget to add uploaded nodes to Node Group so that you are able to see messages.

3. Synchronize the Node Banks using `opccfgupld` and `opccfgdwn`. M1 gets the entries of M2, M2 gets the entries of M1 including their Core IDs.

4. Go to the OVO Application Bank and call the Update Trusts application to update the locally root certificates:

   Certificate Tools -> Update Trusts

   On each management server, select all required managed nodes and execute the application. The agents contact their certificate server and ask for new root certificates.

   You can verify this on all managed nodes by executing command:

   `ovcert -list`

   Two trust certificates should be displayed.

NOTE

You can also trigger this action on the managed node by executing:

`ovcert -updatetrusted`

NOTE

The certificate server is identical to the management server in this scenario.

5. Create or enhance the responsible manager policy on both servers and deploy it to their own agents.
Certificate Handling for a Second OVO Management Server

Assume the second OVO management server has its own Certificate Authority and is used as a backup management server or competence center. Assume that server M1 owns the agents AM1 and that the server M2 initially has no agents.

1. Synchronize the trusted certificates on the management servers: M1 gets the root certificates of M2 and M2 the root certificate of M1.
   a. On OVO management server M1, enter the command:
      \texttt{ovcert -exporttrusted -ovrg server -file <my\_file>}
   b. Copy \texttt{<my\_file>} to the management server M2, for example using \texttt{ftp}.
   c. Enter the following command on M2:
      \texttt{ovcert -importtrusted -ovrg server -file <my\_file>}
   d. Repeat the procedure for management server M2.
   e. To verify that M1 and M2 have the root certificate of the other, on both management server systems, execute the command:
      \texttt{ovcert -list}
   Two trusted certificates should be listed.

2. Configure other management server as regular nodes in the OVO node bank. M1 must be added to the node bank of M2 with its coreid and M2 must be added to the node bank of M1 with its coreid.
   a. Add node M1 in the node bank of M2 and M2 in the node bank of M1 as follows:
      In the Motif Administrator's GUI, select:
      \texttt{Action -> Node -> Add}
      Note: You can also use the command line tool:
      On node M1, enter the command:
      \texttt{opcnodemod -add_node node_name=<M2> net_type=<network\_type> mach_type=<machine\_type> group_name=<node\_group\_name> \}

On M2, enter the command:

```
opcnnode -add_node node_name=<M2> \
net_type=<network_type> mach_type=<machine_type> \
group_name=<node_group_name>
```

b. M1's coreid must be stored in M2's database:

On M1, call the `ovcoreid` command to display the coreid of M1:

```
ovcoreid
```

Note down the displayed value.

On M2, call the `opcnode` command to add M1's coreid into M2's database:

```
opcnnode -chg_id node_name=<M1> id=<core_id_of_M1>
```

c. M2's coreid must be stored in M1's database:

On M2, call the `ovcoreid` command to get coreid of M2:

```
ovcoreid
```

Note down the displayed value.

On M1, call the `opcnode` command to add M2's coreid into M1's database:

```
opcnnode -chg_id node_name=<M2> id=<core_id_of_M2>
```

You can verify that the nodes have been correctly added to the databases by executing the following commands:

a. On M1, enter the command:

```
opcnnode -list_id node_list=<M2>
```

The coreid of node M2 should be displayed.

b. On M2, enter the command:

```
opcnnode -list_id node_list=<M1>
```

The coreid of node M1 should be displayed.

---

**NOTE**

Do not forget to add uploaded nodes to Node Group so that you are able to see messages.
3. Synchronize the Node Banks using `opccfgupld` and `opccfgdwn`. Now M2 receives all agents of M1 and M1 loads the local agent of M2, if not already present in the database.

4. Go to the Application Desktop and call the Update Trusts application to update the root certificate on M1.

Certificate Tools -> Update Trusts

On M1 select AM1, and execute the application. The agent contacts its certificate server and asks for a new root certificate.

---

**NOTE**

You can also trigger this action on the managed node by executing:

`ovcert -updatetrusted`

---

**NOTE**

The certificate server is identical to the management server in this scenario.

---

5. Create or enhance the responsible manager policy on both servers and deploy it to their own agents. M1 must deploy a responsible manager policy to all its managed nodes, in this case, they are M1 and AM1. M2 must deploy a responsible manager policy to its local agent if it was not already a part of M1's environment.

### Using a shared CA in MoM Environments

The scenarios described above show how to merge environments with separate Certificate Authorities. It is also possible to work with only one Certificate Authority. However, this should be considered before setting up an OVO MoM Managed environment.

A disadvantage of sharing one Certificate Authority can be that every agent needs a communication route to this one certificate server, if you want agents to be able to request their certificates at installation time, or later, when further root certificates should be installed on the agent system.

In addition, consider that all OVO management servers and their managed nodes are dependent on one Certificate Authority.
NOTE

A shared Certificate Authority is not the recommended configuration. Using trusts, as explained above, is preferred.

Assume that server M1 has a Certificate Authority and M2 should not have one.

Execute the following steps:

1. Immediately after the installation of M2, remove the local certificates with the following commands:
   
   `ovcert -remove <cert_id>`
   
   `ovcert -remove -ovrg server <cert_id>`

2. Add M2 to the node bank of M1:
   
   On node M1, using the Administrator's GUI:
   
   Action -> Node -> Add
   
   Note: You can also use the command line tool:
   
   On node M1, enter the command:
   
   `opcnode -add_node node_name=<M2> net_type=<network_type> mach_type=<machine_type> group_name=<node_group_name>`

3. Create a certificate for M2 on M1 with the following commands:
   
   `opccsacm -issue -name <M2> -coreid <core_ID_M2> -file <M2_cert> -pass <password>`

   **NOTE**
   
   To display the core ID of M2, on the M1 system, enter the command:
   
   `ovcoreid -ovrg server`

   `opccsacm` also adds the core ID of M2 to the database.

4. Copy the certificate to M2 (HA server) and install it as the server certificate:
   
   `ovcert -importcert -ovrg server -file <my_cert> -pass <password>`
If M2 is not an OVO HA cluster server, call the same command as above but without the resource group server option to install a node certificate:

```bash
ovcert -importcert -file <my_cert> -pass <password>
```

If M2 is an HA system, create an extra node certificate for each physical node. On M1 call:

```bash
opccsacm -issue -name <hostname_M2_cluster_node> -coreid <OvCoreId_M2_cluster_node> -file <my_cert> -pass <password>
```

Copy the node certificates to the M2 cluster nodes and install using the command:

```bash
ovcert -importcert -file <my_cert> -pass <password>
```

5. Instruct every managed node which will be installed by M2 that its certificate server is M1 by placing an entry into the bbc_inst_defaults file. This file is used to automatically generate profiles for the agent installation. The location of the file is:

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

---

**NOTE**

If this file does not exist, create it now using the following sample file as a template:

```
/etc/opt/OV/share/conf/OpC/mgmt_sv/bbc_inst_defaults.sampl
```

Add the namespace and certificate server specifications to your bbc_inst_defaults file as follows:

```
[sec.cm.client]
CERTIFICATE_SERVER <hostname_M1>
```

For the local agent on M2 call:

```bash
ovconfchg -ns sec.cm.client -set \\
CERTIFICATE_SERVER <hostname_M1>
```

6. Unregister the Certificate Server (ovcs) component from system M2 using the command:

```bash
ovcreg -del ovcs
```
7. Create or enhance the responsible manager policy on both servers and deploy it to their own agents. M1 must deploy a responsible manager policy to all of its agents which are to be managed by M2. M2 must deploy a responsible manager policy to its local agent, if it was not already a part of M1's environment.

8. Download the node bank configuration on M1 and upload to M2 by using the `opccfgupld` and `opccfgdwn` tools.
Remote Action Authorization

From the point of view of security, remote actions are a very special case in OVO managed environments. It must be ensured that it is not possible to send a faked remote action to a management server that is then executed on the specified remote system in the environment. In particular, this is sensitive since it is not possible to regard any managed system as a secured system. It is assumed that root access to a managed node is available to unauthorized users.

In addition, one OVO management server of a service provider must be able to manage the environments of several of its customers, while ensuring that no system located in one customer segment is allowed to trigger any actions in any other customer segment.

OVO ensures that action strings, for example, a specific command, cannot be tampered with by a malicious user. On the OVO management server, it is possible to configure:

- On the which systems the OVO management server is allowed to execute an action.
- Whether only "signed actions" originating from an HTTPS agent are accepted.

Action requests contained in OVO messages which specify a target node for the action other than the sender of the message are remote actions and must be handled securely. These remote actions are subjected to additional security checks describe in the following section. Remote actions are only be executed if they pass these security checks.

Server Configuration of Remote Action Authorization

The message manager uses a file-based configuration on the OVO management server to specify authorization of remote actions. The configuration contains a trust section that defines which systems are trusted as action signers, and a list of rules, each of which consist of a condition and an action. Each action request is checked against all condition in the order of their definition. If a condition matches, processing of the action request the action is stopped.
The conditions allow checking properties on an action request, such as source node, target node, or signature. There are only two possible actions: allow and deny. An allow action means that the action request is authorized. A deny action means that the action request is rejected.

Authorization data is logged with the reason for denying authorization. If an action is unauthorized, it is automatically deleted from the message and details about the match and the signature status are added as an annotation to the message. Unauthorized messages never appear in the GUI and therefore cannot be accidentally executed.

Source and target nodes are matched against node groups or single nodes. A dedicated keyword can be used for the management server.

If the new configuration file is missing or contains no rules, all remote actions are disabled. A default configuration file that contains the OvCoreId of the management server is installed with the product. The default configuration file also contains some examples in comments.

During startup, the message manager reads the file:

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

It may also be triggered at runtime to re-read the file.

The syntax of the configuration file is XML based, and according with the following schema:

Figure 3-3  Remote Action Configuration File Syntax
### Remote Action Configuration File Components

<table>
<thead>
<tr>
<th>Elements</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>config</td>
<td>config consists of a trust element and of a list of rule elements.</td>
</tr>
<tr>
<td>trust</td>
<td>The trust element consists of a list of nodeId element, each containing the OvCoreId of a trusted node.</td>
</tr>
</tbody>
</table>
| rule     | Each rule consists of the following components:  
- doc (optional) containing a description. string  
- if (optional) containing a condition.  
- An allow or a deny action.  
The allow and deny actions are empty and define if action execution is allowed or denied. |
| condition| A condition consists of a sequence of optional checks. A condition matches only if all contained checks match. If no check is defined, or if no condition is defined, a match is always successful.  
The checks are:  
- source  
- target  
- certified |
Remote Action Authorization

Table 3-1  Remote Action Configuration File Components (Continued)

<table>
<thead>
<tr>
<th>Elements</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>source</td>
<td>Used to check the source node of an action request.</td>
</tr>
<tr>
<td>target</td>
<td>Used to check against the target node of an action request. Both source and target consist of a set of choices. These checks match if any of the elements match.</td>
</tr>
<tr>
<td></td>
<td>• nodegroup</td>
</tr>
<tr>
<td></td>
<td>The nodegroup element contains the name of a node group from the OVO database. It matches if the request's node is a member of that node group.</td>
</tr>
<tr>
<td></td>
<td>• nodeId</td>
</tr>
<tr>
<td></td>
<td>The nodeId element contains an OvCoreId. It will mach if this OvCoreId is the ID of the request's node.</td>
</tr>
<tr>
<td></td>
<td>• mgmtsrv</td>
</tr>
<tr>
<td></td>
<td>The mgmtsrv element is empty. It matches if the request's node is the management server.</td>
</tr>
<tr>
<td></td>
<td>• nodeAddress</td>
</tr>
<tr>
<td></td>
<td>• nodename</td>
</tr>
<tr>
<td>certified</td>
<td>The certified check allows the values valid and invalid. Valid matches only if a signature and a certificate are provided, with the signature being signed by the certificate's owner, and when the OvCoreId of the certificate's subject is listed in the trust element. Invalid matches all other cases.</td>
</tr>
</tbody>
</table>
The following is an example of a remote action configuration:

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

OVO processes normally run under user root on UNIX systems and under the System account on Windows systems. The root/administrative privileges enable the processes to:

- Access OpenView resources. OpenView files are normally also restricted to privileged access only.
- Allow a switch user for application specific access rights.
- Directly access operating system resources such as log files and configuration files.
- Start application or operating system specific commands and executables.

There may be systems within IT environments that are highly security sensitive and it is necessary to limit the number of processes that have full root permissions to a small, well defined and tested group. In addition, it is desirable to be able to identify the precise process that manipulated critical system resources. This is not possible if many applications are running under the privileged user.

NOTE ovswitchuser is not supported by the OVO HTTPS agent on Windows platforms.

OVO software on UNIX managed node systems can be configured to run under a user that does not have full root permissions, often referred to as “running as non-root”. To run an agent as non-root, access to non-OpenView files and executables must be specifically given to the OVO processes on the node.

All OVO HTTPS agents on UNIX systems can be configured to run under a user other than root using the ovswitchuser tool.
The `ovswitchuser` tool allows the UNIX HTTPS agent on an OVO managed node to run under a user other than the privileged root user. The `ovswitchuser` tool makes the following changes:

- Perform change group ownership on:
  - All registered files of all installed component packages.
  - All files and directories of `<OVDataDir>` recursively.

- Change operating system daemon/service registration to start OVO processes under the new user.

Limitations of Running OVO Agents Under Alternative Users

Agents running under alternative users have the following limitations:

---

**WARNING**

The OVO management server processes must always run under the user `root`. The `ovswitchuser` tool must not be called on the OVO management server system.

---

**NOTE**

The OVO HTTPS agent on Windows platforms does not support `ovswitchuser`. Windows systems must run under the `System` user and cannot be switched to any other user.

---

- Actions can only be executed if the account under which the agent runs has suitable privileges.
- It is not possible to access files or any other operating system resources unless the agent account has suitable privileges.

---

**NOTE**

It is possible to circumvent access restrictions by implementing a `sudo` program, which gives the agent user additional capabilities for specific operations. For further details, refer to “Working with Sudo Programs on UNIX Agents” on page 81.
Configuring an Agent to Run Under an Alternative User

Preparing the System Environment

WARNING

Do not use ovswitchuser.sh on the OVO management server system. The OVO agent on the OVO management server must run under the user root.

NOTE

After the change of user has been made using the ovswitchuser command, the agent processes must be run under this newly assigned user and no longer under the user root.

For HTTPS agents, you must select a UNIX group for the agent. All users under which the agent is to run must belong to this group. If you are migrating from a non-root DCE agent to a non-root HTTPS agent there are some issues to consider. For example, if the DCE non-root agent is run as user OVO_Agent of group Security. No-one except user OVO_Agent or the super-user is able to read runtime files of this agent. With the HTTPS agent, permissions are defined and granted at the group level and all users belonging to the group Security can access the runtime data of the agent. Therefore, it might be necessary to create a new group Security2 and put the user OVO_Agent into the group Security2. Otherwise all other users in the group Security could access the runtime data of the agent, including private keys.

NOTE

The users and groups used in the above scenario are only examples. You are free to choose your own user and group names.

As long as the DCE agent user belongs to a group containing only trusted users, when the DCE agent is replaced by an HTTPS agent which should also run as non-root, no migration step is needed. The HTTPS agent can run under the same user that was used for the DCE agent.
umask Setting on UNIX  The non-root concept relies on the user under which the agent runs belonging to a specific UNIX group. Therefore the group bits of any files that are created by OV applications must be set. This allows OV applications to be run under dedicated users if required, while sharing the same resources, for example log files. Therefore, it is recommended to set the umask to suit the users that are used to run OV applications.

A umask setting of 02 is preferable. 022 would cause problems when multiple applications are run under different users.

If only the OVO agent is installed or if all applications run under the same user, the umask does not need to be set.

Installing an Agent Using an Alternative User on UNIX Managed Nodes

Complete the following steps to run a managed node under an alternative account to root:

1. Install the agent software on the desired node as usual.
2. Stop the agent with the command:
   
   opcagt -kill

   **NOTE**
   Do not use the command:

   opcagt -stop

   This stops the agent processes but not the core OpenView processes. When you later start the agent processes with the command:

   opcagt -start

   As the core processes are already running under the root user, all other process are also started under the root user.

3. Set the umask of the user to grant Group Permissions.
4. Call the ovswitchuser command:

   /opt/OV/bin/ovswitchuser.sh -existinguser <my_user> \
   -existinggroup <my_trusted_group>
By default the OVO HTTPS agent uses port 383 for network communication. This is a privileged port which can only be opened by user root. Therefore, you must select one of the following alternatives to configure the non-root agent to communicate over the network:

5. You must configure the port that is to be used.

If you want to continue using the reserved, privileged port 383, set the SUID bit as described in the first point below. However, if you wish to use an alternative port, rest it using the following ovconfchg command as described in the second point.

---

**WARNING**

Only apply one of the following approaches: `setuid` OR change the `PORTS` setting.

---

- It is possible to continue using the reserved, privileged port 383 by setting the SUID bit on the communication broker executable. Then, the communication broker only uses root privileges to open up the port and then switches back to the agent user for all other activities.

  Set the `setuid` bit of the `ovbbcccb` binary with the following command:

  ```
  chmod 4550 /opt/OV/bin/ovbbcccb
  ```

- Select a non-privileged `ovbbcccb` port. Change the port from 383 to a desired port with a value greater than 1024.

  For HTTPS agents, the communication broker port on a system where the HTTPS agent is not running under user root is changed to a non-privileged port. As a result, all other applications using the communication broker on this node experience the same limitation. If you want to use an alternative port, refer to “Configuring the OVO Management Server For Agents Running Under Alternative Users”.

  On a managed node, use the command:

  ```
  ovconfchg -ns bbc.cb.ports -set PORTS \<FULL_DNS_NODE_NAME>:<NEW_PORT_NUMBER>
  ```
Configuring the OVO Management Server For Agents Running Under Alternative Users

If you use a different port than the default 383 on a managed node, you must also configure this on the OVO management server. In addition, the port to be used for a particular node must be known to all OVO management servers that need to contact that managed node. This is done by setting the `bbc.cb.ports PORTS` variable on OVO management servers.

For example, let us assume that we have a managed node with hostname `ovo_node.sales.mycom.com`, the OVO management server hostname is `ovo_srv.sales.mycom.com`. The new `ovbbccb` port on `ovo_node.sales.mycom.com` is 8001.

This port value must be set on the managed node and the OVO management server.

To set an alternative value for the `ovbbccb` port, enter the following command on both OVO management server and the managed node:

```
ovconfchg -ns bbc.cb.ports -set PORTS \
"ovo_node.sales.mycom.com:8001"
```

To individually set the new port values for each managed node is inefficient and error-prone. Wildcards are recognized and should be used to specify groups of managed nodes as used in the following examples.

Let us now assume that all nodes of domain `sales.mycom.com` should use port 8001. To set this port for all systems in this domain, enter the following command on both OVO management server and the managed nodes:

```
ovconfchg -ns bbc.cb.ports -set PORTS \
"*.sales.mycom.com:8001"
```

However, it is recommended that OVO management servers always use port 383. So we should modify the previous step and enter the following command on both OVO management server and the managed nodes:

```
ovconfchg -ns bbc.cb.ports -set PORTS \
"ovo_srv.sales.mycom.com:383,*.sales.mycom.com:8001"
```

It is important that the `bbc.cb.ports PORTS` entries on OVO management servers is always up-to-date. It is not normally important for a managed node to know which port is used by another managed
node. Therefore, only the setting on the OVO management server and the setting on a newly installed managed node agent must be considered. No update of the PORTS setting on existing agents is needed.

**Changing the Default Port**

It is recommended that you maintain the PORTS setting in a central place on the OVO management server system and use wildcards to reduce the need to make changes on the management server.

A sample configuration file with examples of how to set up parameters is available:

```
/etc/opt/OV/share/conf/OpC/mgmt_sv/bbc_inst_defaults.sampl
```

Take a copy of the `bbc_inst_defaults.sampl`, rename it `bbc_inst_defaults`, and modify it as follows:

Make a `bbc_inst_defaults` file entry of the form:

```
[bbc.cb.ports]
PORTS = ovo_srv.sales.mycom.com:383,*sales.mycom.com:8001
```

As a result, all newly installed agents are automatically provided with the information that `ovo_srv.sales.mycom.com` uses port 383, while all agents matching `*.sales.mycom.com` use port 8001. The `bbc_inst_defaults` file is the basis for the “Agent Profile”, which is installed with every new managed node. The “Agent Profile” is explained in more detail on page 77.

If a new managed node system belongs to the domain `*.sales.mycom.com`, the OVO management server is correctly configured and port 8001 is used. You can check this by entering the following command on the OVO management server:

```
ovconfget bbc.cb.ports
```

If the OVO management server does not have the correct settings, take the value from the `bbc_inst_defaults` file and call `ovconfchg` to update the OVO server with a command of the following form:

```
ovconfchg -ns bbc.cb.ports -set PORTS "<ovo_server>:383,<system1>:<port1>,<system2>:<port2>, *
*.<domain1>:<port3>,*.<domain2>:<port4>"
```
Agent Profile

An agent profile maintained on the OVO is a list of configuration settings which is copied to the agent at install time. The profile contains some default values do not need to be configured in the `bbc_inst_defaults` file. Any settings defined in the `bbc_inst_defaults` file are also added to the agent profile.

The profile is concerned in ALL types of agent initial installations.

Use of the `bbc_inst_defaults` file is optional. If it exists, it is processed and the agent profile is enriched with data from `bbc_inst_defaults` file.

In case of manual agent installation, you can create the agent profile using the command:

```
/opt/OV/bin/OpC/opcs -create_inst_info <node>
```

The profile is located at:

```
/var/opt/OV/share/tmp/OpC/distrib/<hex_IP_addr_of_node>.i
```

**NOTE**

When `opcs` is called, it prints the `<hex_IP_addr_of_node>` to stdout.

Copy the profile together with the software packages to the node and enter a command of the following form:

```
opc_inst  -config <profile_name> ...
```

The utility `opcs` includes the option:

```
create_inst_info
```

If you call `opcs -create_inst_info <node_specifier>` a file is created at:

```
/var/opt/OV/share/tmp/OpC/distrib/<hex_IP_addr>.i
```

for each node from `<node_specifier>`.

This file contains the installation defaults for the node with IP address `<hex_IP_addr>`. The file is automatically copied to the target node during remote agent installation using `inst.sh`, or you can use it for manual agent installation.
The `opcsw -create_inst_info` command creates agent profiles using configuration data from the file:

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

on the management server and the following additional information from the OVO database:
• **CORE_ID**: OVCoreID of managed node. An optional parameter which is added to the profile if a value for CORE_ID is available in the OVO database under the namespace sec.core. If the CORE_ID parameter is not present in the database nor on the managed node, one is automatically created on the agent.

• **MANAGER**: Long hostname of primary OVO management server in namespace sec.auth.

  Only node MANAGER is authorized to perform config-, deployment-, message-, or action-execution related tasks after initial installation.

• **MANAGER_ID**: OVCoreID of MANAGER in namespace sec.auth.

  MANAGER_ID corresponds to MANAGER and is needed to perform the authorization checks.

• **CERTIFICATE_SERVER**: Long hostname of the system where a certificate request is issued (certificate authority) in namespace sec.cm.client.

  If no valid node certificate is present on the managed node, one is requested from CERTIFICATE_SERVER using the CORE_ID as the identifier.

• **PROXY**

  Defines which proxy and port to use for a specified hostname.

These five parameters are the minimum initial settings required on a managed node. It is possible to overwrite them in the bbc_inst_defaults file, for example, if you have one dedicated certificate authority for several OVO management servers.
Upgrading and Patching an Agent Running Under an Alternative User

Upgrading and patching DCE/NCS agents requires you to call \texttt{opcs\text{\char46}switchuser} after each agent software installation, including upgrades and patch installations. This modifies the ownership of all OV files and directories to the customer defined owner. Additionally it changes the startup script to start the OVO processes under this specific user. \texttt{opcs\text{\char46}switchuser} must be run very time you install additional OpenView modules to a specific system so that the ownership of the new files is changed to match the non-root user.

Running \texttt{ovswitchuser} is not required for upgrading and patching HTTPS agents. How to handle upgrading and patching HTTPS agents is described in the following sections.

Copy To Node and Manually Install Later

\textbf{NOTE}

The “Copy To Node and Manually Install Later” concept is only valid for HTTPS nodes.

It is possible that an OVO administrator does not have root access to a system and the OVO agent is running as a non-root user. However, for HTTPS agents, if the communication broker is running on a node, you do not need to enter passwords, as data transfer works without them. Without root access, the complete remote installation of the agent, as described in section “Installing Agents Manually” on page 118, cannot be performed. It is only possible to copy the agent packages to the managed node system and a manual installation must be done at the node system itself. Native installer calls, such as \texttt{pkg\text{\char46}add} on Solaris, \texttt{rpm} on Linux, \texttt{swinstall} on HP-UX, need superuser privileges. This HTTPS node concept can be viewed as “copy to node and manually install later”.

If you run a non-root agent and you want to deploy a sub agent, a patch or complete upgrade package which requires native installer access, the following is done automatically:

1. The bits are copied to \texttt{/tmp/<pkg\_name>}.

2. The installation cannot proceed further, because the deployer is not able to call a native installer as this requires root capabilities.
It finishes with OK but generates a warning message.

3. Inform an authorized person on the target managed node that the packages are locally available. This administrator can then continue with the installation by calling the opc_inst script in the same way as for a manual agent installation.

**NOTE** HTTPS-transfer is preferred to bootstrap transport methods. This means that a remote sub-agent, patch or upgrade installation of a non-root agent will not ask for passwords but on the other hand it will terminate after copying the bits. You are not prompted for the root password and the installation must be triggered explicitly. However, the additional manual installation step respects the current agent user.

**Working with Sudo Programs on UNIX Agents**

**NOTE** The “Copy To Node and Manually Install Later” concept and the use of sudo programs is only valid for HTTPS nodes.

One way to get the required rights is to configure a tool like sudo and configure the OV_SUDO setting. Sudo allows a permitted user to execute a command as the superuser or another user, as specified in the sudoers file. The real and effective uid and gid are set to match those of the target user as specified in the passwd file. The group vector is also initialized when the target user is not root. By default, sudo requires that users authenticate themselves with a password. By default this is the user's password, and not the root password. After a user has been authenticated, a timestamp is updated and the user may then use sudo without a password for a short period of time. By default, 15 minutes unless overridden in the sudoers file.

**NOTE** Sudo is free software and it is distributed under BSD-style licence. It can be obtained from http://www.sudo.ws. Sudo software is not packaged as part of the OVO software.
Let us take an HTTPS agent running on a Solaris managed node as a non-root user, ovo_user.

The procedure is as follows:

1. Open the /etc/sudoers file.
2. Add the following line into /etc/sudoers file. Use vi /etc/sudoers or visudo command.
   
   ovo_user ALL=(root) NOPASSWD: /var/opt/OV/
   installation/incoming/bundles/OVO-Client/opc_inst
   
   Only the installation script opc_inst is called under a superuser, root.

---

**NOTE**

This command is valid for remote installation using the Administrator IU or using opc_inst. In all other cases, the actual path for opc_inst must be substituted.

---

If NOPASSWD is not specified, you should enter your own password, for example for the user ovo_user, and not superuser (root) password.

**How to Setup a Sudo Program**

---

**NOTE**

The bootstrap installation does not support OV_SUDO.

---

OpenView installation utilities that make native installer calls contain code of the form:

$ {OV_SUDO} opc_init

If the OV_SUDO variable is not set, it is interpreted as an empty string and ignored.

If the OV_SUDO variable is set, the variable is either exported from the non-root user’s login shell, or it is read using ovconfget ctrl.sudo and then added to the environment by the install scripts.
NOTE
Reading the OV_SUDO variable using ovconfget ctrl.sudo has higher priority than exporting its value from the non-root user's login shell.

A typical bootstrap installation of a non-root agent with sudo requires the following steps:

- Install agent as root.
- Call /opt/OV/bin/ovswitchuser to set the preferred user and group.
- Set preferred sudo program using the command:
  
  ovconfchgc -ns ctrl.sudo -set OV_SUDO \
  <my_sudo_with_full_path>

- Set preferred sudo user using the command:
  
  ovconfchgc -ns ctrl.sudo -set OV_SUDO_USER <my_sudo_user>

- Set preferred sudo group using the command:
  
  ovconfchgc -ns ctrl.sudo -set OV_SUDO_GROUP <my_sudo_group>

NOTE
The benefit of setting a sudo allows automatic sub-agent, patch and upgrade installation of non-root environments without entering passwords. Conversely, a remote bootstrap installation requires that an OVO administrator knows a super-user password of the managed node.

The remote agent installation first checks as which user an agent is running and whether OV_SUDO is setup. It decides then, whether “copy to node and manual install later” is needed. Depending on this bootstrap installation with password prompting or automatic installation is chosen.
A Comparison of DCE and HTTPS Alternative User Concepts

All OVO agents on UNIX systems can be configured to run under a user other than root. This is done using the opcswitchuser tool for DCE- and NCS-based UNIX agents, and the ovswitchuser tool for the HTTPS agent.

For a DCE/NCS agent, all files and directories of any OV application are set to the same user and group by the opcswitchuser tool.

For DCE/NCS nodes:

```
/opt/OV/bin/utils/opcswitchuser.sh <my_trusted_user> <my_group>
```

**NOTE**

opcswitchuser.sh is not located in `/opt/OV` on all platforms. Check the actual value of `OVInstallDir` and `OVDataDir`.

- You must select a UNIX group for the for the user under which the HTTPS agent is to run. This is not necessary for the DCE/NCS agent. For more details, refer to “Preparing the System Environment” on page 72.

- The HTTPS agent has file access rights opened for the assigned user and all other users which belong to the same group as the user of the HTTPS agent. The DCE/NCS agent can only be run under the assigned user. Example: OVO queue files: HTTPS 0660, DCE 0600.

**NOTE**

Before changing the user under which the agent processes are to be run, set the `umask` of the user to grant Group Permissions and shutdown the agent.

- The HTTPS agent has the `group-id` bit set on its base directories.
  
The `group-id` bit guarantees that all files created under such directories will belong to the agent's group. This also works if the primary group of the user under which the agent is running is different from the group of the agent files and directories.
For example, the primary group of user OVO_Agent is Security, agent files and directories belong to group Security2. Now also add OVO_Agent to group Security2 (Security remains the primary group of OVO_Agent) and run the agent under user OVO_Agent. All files created by the agent running under the user OVO_Agent will belong to Security2. This mechanism allows OV components to run under different users but share common files.

- The set group-id bit may cause warnings of security check tools like medusa, which can be safely ignored. On DCE/NCS agents, no such warnings occur.

- No “copy to node and manual install later” concept for DCE/NCS nodes.

- No sudo concept for DCE/NCS nodes.

- For DCE/NCS nodes it is necessary to call opcswitchuser after each patch/upgrade installation on non-root agent. On HTTPS agent this is not required. You call ovswitchuser only once after bootstrap installation. Later you call ovswitchuser only, when you want to change the group/user of the agent, for example, back to root.
4 Concepts of Managing HTTPS Nodes
Controlling HTTPS Nodes

The OVO management server can perform the following functions on HTTPS nodes:

- Remote control of HTTPS agents.
- Remote and manual installation of HTTPS agents.
- Remote and manual patch installation and agent upgrade.
- Remote and manual configuration deployment.
- Support of multiple parallel configuration servers for HTTPS agents.
- Heartbeat polling.
- Security management of HTTPS nodes.
- Support of HTTPS nodes through the OVO management server APIs and utilities.

The following sections explain some new concepts for HTTPS nodes.

- “Configuration Deployment to HTTPS Nodes” on page 89
- “Heartbeat Polling of HTTPS Nodes” on page 96
- “Remote Control of HTTPS Nodes” on page 98
- “OVO Server Components and Processes” on page 29
Configuration Deployment to HTTPS Nodes

Configuration deployment to HTTPS agents differs slightly from that of DCE-based nodes:

- Policies are used by HTTPS agents in place of Templates.
- Instrumentation is the single term used by HTTPS agents for Actions, Commands, and Monitors.
- A configuration parameter schema with a name-value pair policy type for HTTPS agents replaces nodeinfo and opcinfo files.
- mgrconf file is enhanced for HTTPS agents by a role model-based security authorization mechanism.

**NOTE**

The same responsible manager file can be used to support both HTTPS and DCE managed nodes.

However, a responsible manager file may be created for HTTPS nodes only:

```
/etc/opt/OV/share/conf/OpC/mgmt_sv/respmgrs/allnodes.bbc
```

If it exists, it has a higher priority than the allnodes file but lower than the <hex_IP_addr> files for an HTTPS node. The file is distributed automatically together with policies, in the same way as the allnodes file with templates.

allnodes.bbc can be empty or contain only a subset of the settings from the allnodes file. An empty allnodes.bbc file means that no MoM configuration is distributed to an HTTPS node. All OVO management server systems specified in a responsible manager file going to an HTTPS node must use HTTPS as the communication type and have an OvCoreId.

The following sections explain the new configuration management concepts introduced with the HTTPS agents.
Policy Management

A policy is a template in XML format, with the strict separation of data and meta information. The header contains attributes such as name, type, version, and state. Five operations are possible on policies: install, remove, enable, disable and list. Template files contain all individual templates of a certain source type in one file, a policy file contains only the content of one template and this information is referred to as the policy data.

It is possible to manually install and remove policies using the `ovpolicy` tool, provided that you adhere to some guidelines.

Existing OVO templates can also be used with HTTPS agents as these are converted into policies at distribution time by the `opcbbcdist` process. The `mgrconf` and the `nodeinfo` configuration types are now treated as policies. Only one `mgrconf` file and one `nodeinfo` file are required, and a unique policy id is used.

In addition to the unique policy id, the header contains the policy name, policy type name, policy version, policy type version, and status. These attributes are generated by `opcbbcdist` as the data is being deployed.

Only one version of a policy can be installed on a node. A policy is identified by its id, but also the name plus policy type must be unique.

All policies that are deployed from the OVO server are allocated the version number 1 as OVO does not support policy versioning.

The status of a policy deployed for the first time is set to enabled. If the policy is already present on the system, a newly deployed policy assumes the status of the policy it replaces.

There is a utility called `opctemplate` for HTTPS nodes, which is wrapper for `ovpolicy` and allows common definitions with DCE nodes, for example in the application desktop.

Instrumentation Management

On HTTPS nodes, the actions-, commands-, and monitor directories are replaced with:

```
$OVDataDir/bin/instrumentation
```

which can have one level of sub directories. All instrumentation programs are installed at this location.
NOTE

The directory for executables on the OVO management server is located under:

/var/opt/OV/share/databases

No instrumentation directory is created and the directories actions, commands, and monitors are used.

NOTE

Typically, action, command, and monitor executables are referenced in OVO templates. As long as these executables are not referred with their full path in policies, this change is transparent, because the new locations of the binaries is also added to the path variables of utilities like the OVO action agent, monitor agent and logfile encapsulator.

Files from the monitor directory on the OVO management server are installed on the agent with the rights 744, all others with the rights 755. This is identical to the settings on DCE-based nodes.

The configuration management process can also update running executables. Scripts and binaries of running executables are renamed and allowed to complete their tasks. Subsequent execution of these programs use the newly installed files.

Manual Installation of Policies and Instrumentation

It is not possible to copy policy data directly to a managed node because the agent must receive the configuration data in a secured format. This is required to avoid illegal manipulation of configuration data by unauthorized persons on the managed nodes.

The opctmpldwn tool is used to prepare the manual installation of policies on the OVO management server. The output data is stored in a directory on the management server system dedicated to the managed node.

There are minor differences between how opctmpldwn handles DCE and HTTPS nodes:
For HTTPS nodes, the nodeinfo and mgrconf data are regarded as policies and therefore contained in the directory mentioned above. For DCE-based node, the nodeinfo data is disregarded.

Templates and policies are secured using different methods. A template is encrypted with a node-specific key. The policy data is signed through a management server specific certificate while a policy header is only secured through file rights.

**HTTPS Agent Distribution Manager**

opcbbcdist is the configuration management adapter between the OVO management server and the HTTPS agents. Its main function are:

- Convert templates into policies.
- Create instrumentation from existing actions, commands, and monitors.
- Convert ECS templates into policies and their associated circuits.
- Switch nodeinfo settings into the XPL format used on HTTPS nodes.

Opcbbcdist is the counter part of opcdistm, the distribution manager for all other communication types. Just like opcdistm, it uses the internal file system interface:

/var/opt/OV/share/tmp/OpC/distrib

to get the information about what data should be deployed. Opcbbcdist also distinguishes between the four configuration categories:

- Policies/templates
- Instrumentation actions/commands/monitors
- nodeinfo
- mgrconf

Unlike opcdistm, opcbbcdist only accepts requests from other OVO management server components of the form deploy configuration types xyz to node abc. These requests may be issued by the GUI, by a configuration API or by opcragt -update and opcragt -distrib.

opcbbcdist possesses an automatic retry mechanism which is started if it was not possible to reach a node and new data is present for it. You can also manually trigger a retry by calling opcragt -update.
When `opcbbcdist` or `opcdistm` complete a task for a certain node, you get a message in the browser confirming correct distribution of configuration data. If tasks are not completed, messages, such as `Node Unreachable`, are displayed.

`Opcbbcdist` transfers instrumentation data first, then policies. This is done to avoid synchronization issues when an executable is referenced in a template. In addition, `opcbbcdist` follows a simple transaction model: only if all data of a certain configuration type is successfully deployed, is the next category processed. The distribution of one configuration type is regarded as one transaction. If a transaction fails, it is rolled back and retried later. This schema is also applied when `opcbbcdist` is stopped due to OVO server shutdown.

**Configuration Push**

The OVO management server triggers all configuration deployment tasks to HTTPS nodes. The OVO server pushes configuration data down to the agent and there is only out-bound communication. The more secure OVO management server triggers the managed nodes.

A disadvantage is that a managed node must run with old data in the case of the system not being reachable when new configuration was distributed. The OVO management server must poll all nodes for which configuration is present but could not be delivered. The OVO management server does this task:

- at least once an hour per pending node.
- when the server is restarted.
- when the configuration push is explicitly triggered by `opcragt -update, opcragt -distrib`, or within the GUI by pressing the Distribute button, or by directly calling the API associated with the command.

---

**NOTE**

In addition, DCE-based agents ask the OVO distribution manager `opcdistm` for new configuration data after system reboot or agent restart.

---

A monitor called `dist_mon.sh` checks for pending distributions. If any data in the configuration transfer directory:
/var/opt/OV/share/tmp/OpC/distrib

is older than 30 minutes, a message is displayed that specifies the managed node where a distribution is pending.

**Delta Distribution**

By default in OVO, the distribution process, known as delta-distribution, only deploys data which has been modified or added since the last configuration transfer. This minimizes the amount of data transferred and reduces the number of reconfiguration requests for interceptors and other sub agents. If required, the complete configuration can be re-deployed to the managed node.

In the delta-distribution mode, the OVO management server requests the policy inventory of the managed node and time stamps of the last instrumentation distribution. The policy inventory is compared with the policy assignment list and opcbbcdist computes and executes the required policy removal and installation tasks for the node. For instrumentation deployment, the time stamp of the last deployment is compared with the time stamps in the management server instrumentation directories. All files on the OVO management server that are newer than the corresponding file on the managed node are distributed. No instrumentation data is ever removed from the managed node, except if the opcragt -purge command line command and option is applied. This cannot be executed from the Administrator UI.

**Multiple Parallel Configuration Servers**

Multiple parallel configuration servers are supported for HTTPS nodes. The OpenView policy concept allows multiple OpenView products to independently work with policies on an agent by providing an owner concept for policies. The policy header includes an attribute *owner*, which can be set by the OVO management server. This is a logical association using a concept of agreements between management servers to decide which management server is responsible for which configuration (policies) on an agent.

MoM configurations and nodeinfo are regarded as policies and both contain owner strings in their policy headers. They can only be removed or modified by their owner. Normally all policies (templates) associated with an OVO management server can be modified by this management server. This means that two different management servers will not
interfere when distributing policies to the same agent, because they have a different name. However, there are cases when one management server interferes with policies deployed to the same agent by another management server. When identically named templates are assigned and distributed to the same agent by the second server, the existing instances of those policies are first removed and re-deployed with new owner strings. If you need to, you can also manually overwrite the owner string by using the config setting `OPC_POLICY_OWNER` in the `opc` namespace on the agent. The owner string is:

```
OVO:<server_full_qualified_name>
```
### Heartbeat Polling of HTTPS Nodes

Heartbeat polling of HTTPS nodes and DCE-based nodes is very similar. Heartbeat polling of OVO managed nodes is driven by the OVO request sender process `ovoareqsdr` and is divided into three phases:

- The request sender `ovoareqsdr` sends ping packages to check whether the node is reachable.
- The HTTPS agent communication broker is polled.
- OV Control RPC server is requested.

**NOTE**

You can use the `RPC_only` mode, where the ping phase is omitted, to get through firewalls which have the ICMP filter enabled. In `RPC_only` mode, less checks are executed. Should a problem arise, the detail available from the error messages is reduced.

You can set different polling intervals per node.

HBP error messages of HTTPS nodes and DCE-based nodes are very similar. For example, the message `DCE rpcd is down` for DCE-based agents corresponds to `communication broker is down` for HTTPS agents and is allocated the same error number.

**NOTE**

Heartbeat-polling of HTTPS nodes is done without using SSL to minimize CPU load.

---

**Reduce Network and CPU Load**

Heartbeat polling includes the option `agent_sends_alive_packages`. When enabled, the agent regularly informs the OVO management server that it is working correctly by sending ping packages. The OVO management server only starts polling when it has not received an alive package from one or more managed nodes in the last period.
The server plays an active role only in failure cases and the alive packages are very small. This results in an extreme reduction of network and CPU load. This feature is of great benefit when large environments are managed with no firewalls between managed nodes and the OVO management server.
Remote Control of HTTPS Nodes

The `opcagt` utility is used to control agents from the OVO management server. All supported operations can be simultaneously executed on HTTPS nodes and non-HTTPS nodes. These operations include start, stop, get status, primary manager switch, get and set configuration variables, as well as configuration distribution.

There is a wrapper called `opcagt` on HTTPS nodes. This utility can be used to perform remote control tasks by application launch from the operator's desktop. It allows to setup a common action definition for any kind of OVO managed nodes.

The output format of `opcagt -status` as well as for other `opcagt` operations looks identical for HTTPS nodes and DCE-based nodes. Error messages are also very similar.

Subagents are identified by names on HTTPS nodes and by numbers on DCE nodes. Therefore, you can specify aliases of the form:

```
<alias>  <maps_to>
```

in the configuration file:

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

The entries `1  EA` and `12 CODA` are pre-defined. To automatically transform the `-id 1` into `-id EA` for HTTPS managed nodes, enter the command:

```
opcagt -status -id 1 <BBC_nodes_and_DCE_nodes_list>
```
5 Working with HTTPS Nodes
Configuring HTTPS Nodes

HTTPS nodes are configured in the same way as DCE-RPC- and NCS-RPC-based nodes and configured through the Add, Modify, and Copy Node windows in the OVO Administrator’s user interface or using the opcnode(1m) and the Node Communication Options and Node Advanced Options windows.

As OVO administrator, do the following for HTTPS nodes:

- Specify a new communication type HTTP-Based for supported platforms.
- Specify whether a node’s IP address is static or dynamically assigned using DHCP. See “Managing HTTPS Agents on DHCP Client Systems” on page 151.

**NOTE**

When changing the communication type between DCE and HTTPS, the DCE agent software is automatically removed. Local configuration or runtime data, including opcinfo file settings, ECS data and fact stores, Embedded Performance Component database files, are converted and re-used by the HTTPS agent.

Security of HTTPS communication is achieved using certificates which results in some new steps being required to install HTTPS agents. The steps that you must complete are:

1. Install the OVO HTTPS agent software on the managed node through the Add Node window. The node automatically sends a certificate request to the OVO certificate server which is automatically granted. If auto-grant is disabled, the next two steps are also required.

2. Select the nodes to which you want to grant certificates from the OVO Node Certificate Requests window.

3. Grant the certificate requests to the selected nodes.
   The nodes for which certificates have been granted are added to the Holding Area (default) or in the configured layout group as specified in the configuration setting OPC_CSA_LAYOUT_GROUP in the namespace opc.
Installing OVO Software Automatically on HTTPS Nodes

OVO software installation is controlled from the Add Node window, illustrated in Figure 5-1.

Figure 5-1 Add/Modify Node Window For an HTTPS Node
Working with HTTPS Nodes
Configuring HTTPS Nodes

NOTE
Windows does not have a boot startup system comparable to UNIX. To start ovcd on Windows independent of user login, ovcd is registered as a service. Based on the default START_ON_BOOT value, the installation sets the service startup to Automatic or Manual. However, subsequent changes to the START_ON_BOOT flag have no effect on the ovcd service registration.

On Windows, you must change the service startup manually as follows:

1. Go to Start -> Settings -> Control Panel -> Administrative Tools -> Services
2. Double-click the HP OpenView Ctrl Service and from the General tab of the Properties window, set the required Startup Type.

This behavior can be noticed in the following use cases:

Agent Installation from the GUI
When add the managed node to the Node Bank, you can also select the option Automatically update system resource files in the Add Node window. If you select this option for a Windows node, the ovcd control service is registered with start-up type Automatic, and the agent starts automatically after a reboot. If you do not select this option, the ovcd service is registered with start-up type Manual. In this case, you must manually start the agent after each reboot.

Manual Agent Installation
Using opcactivate, you can specify the -nb option (or an equivalent option) which has the same effect as selecting Automatically update system resource files from the OVO GUI.

Settings selected during the agent installation cannot be changed using OVO. To change these settings, use the Windows Control Panel.

NOTE
The Windows agent install script opc_inst.vbs creates the opc_inst.log log file. Installation steps and results are recorded in this file. While the script is running it resides in %TMP% of the user under which the installation is run. The default is Administrator.

It is copied, after a successful installation, to <OVInstDir>\data\log.
NOTE

You can define settings on the management server, which are deployed to the managed nodes at installation time. Basic parameters, such as communication ports or http proxy settings, that are used by many nodes can be define this way. Common scenarios include:

- Need to install many OVO agents on a subnet or domain. Due to firewall restrictions, the default port of the Communication Broker (383) cannot be used and you want to avoid having to manually set the Communication Broker port on every node during agent installation.
- Configure default settings for installation of managed nodes at a central point as the nodes of a subnet or domain share many settings.
- OVO agents are manually installed on a subnet behind a firewall. Common parts of the installation can be automated.

You can maintain these common settings on the OVO management server using the file:

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

A sample configuration file with examples of how to set up parameters is available at:

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

Take a copy of the bbc_inst_defaults.sampl, rename it bbc_inst Defaults, and modify in accordance with the syntax specified in the sample file.

NOTE

If you want to allocate a specific OvCoreId for a new node, manually add it as follows before starting the agent software installation:

On the OVO management server, enter one of the following commands:

```bash
opcnode -chg_id ... id=<id>
```
or

```bash
opcnode -add-node ... id=<id>
```
During agent installation, the OvCoreId from the OVO database is used for the specified managed node.

This is recommended when re-installing a node managed by many management servers. Re-using the original OvCoreId avoids having to update all the OVO management servers.

When installing certificates manually, everything is prepared on the OVO management server before an agent is installed, including creating an OvCoreId, generate a certificate, add the node with the new OvCoreId to the database. Only after these steps can the agent software be installed on the managed node. Finally the certificate must be copied to the managed node.

**NOTE**

HTTPS agents normally run under the SYSTEM account. If an HTTPS agent is running on an Installation Server, it must have access to other nodes – this is not possible using the account SYSTEM.

To install Windows agent software using an installation server, the OVO agent acting as installation server cannot run as SYSTEM (which is the default). Instead, this agent must run under an identity, which is able to access the target managed node using regular Windows access mechanisms to the admin drive. This is usually either:

- A domain administrator
- Windows pass-through authentication is in place (identical user/password on both nodes). Use the ovswitchuser command to change the identity of the OVO agent acting as installation server to accomplish this.

The ovswitchuser() command is not generally supported by the HTTPS agent for Windows. The HTTPS agent for Windows must run under the system account unless it is being used as an Installation Server.

For further information, refer to “Configuring a Windows Installation Server” on page 109.

To install the OVO software automatically:

1. Open the Add Node window by selecting:

   Actions: Node -> Add

   **Chapter 5**
from the menu bar of the OVO Node Bank window (see Figure 5-1) and enter the following information:

2. Enter a label used to identify the system.

3. Enter the hostname of the system.

4. Use the System acquires IP dynamically (DHCP) checkbox next to the IP address if you want to specify that the IP address of the selected HTTPS node is dynamic. This is most useful when the node uses DHCP to get its IP address. Similarly, if the IP address of a node is changed manually and Dynamic IP is selected, the change is also updated in OVO. If DHCP is selected, OVO automatically deals with managed node IP address changes without causing any problems, without losing any messages or without creating an inconsistent or undefined state.

---

**NOTE**

Dynamic IP is only supported on HTTPS nodes. Dynamic change of hostname is not supported.

---

5. Select the type of managed node. Controlled is the default.

Type of managed node is also accessible from the OVO Node Defaults window.

---

**NOTE**

Automatic actions will execute on HTTPS managed node set to Monitored Only. However, operator-initiated action will not execute on HTTPS managed node set to Monitored Only.

---

**NOTE**

Setting Message Allowed as the node type prevents the distribution of software and instrumentation to that node.

Changing the Type of Managed Node for HTTPS nodes does not distribute a nodeinfo file to the managed node. However, for all node types, changing the type from Controlled to Message Allowed or Disabled, stops all agent processes except ovcd.

---

6. Enter the desired heartbeat polling settings (optional).
7. Select the **Automatic (De-)Installation** option when adding a managed node to the OVO environment (optional).

The communication type and settings to be used by the node are displayed in the **Node Communication Options** window. To access this window, click the **Communication Options...** button on the Add, Modify, or Copy Node window for a node.

**Figure 5-2   Node Communication Options Window**

An HTTPS managed node displays HTTPS as its **Communication Type**. The unique identifier, **OVCoreID**, is displayed for reference.

Switching between communication type HTTPS and another communication type automatically changes the platform for the node and removes all values for this node that are only relevant for the newly selected communication type.

HTTPS is the default for new nodes. SNMP-based automatic agent platform detection for newly-added nodes always selects, if available, the HTTPS-based platform.

When you change a node's platform, all node, communication and advanced options are retained, where necessary. This way, switching a node to HTTPS-based management is simplified by retaining the existing settings and generally maintaining the original monitoring view.
The root directory for installation of the agent software is configurable for the HTTPS agent on Microsoft Windows nodes.

**NOTE**

It is not possible to specify a customized log directory and maximum log size for HTTPS nodes, since the new OpenView file system layout and OpenView logging mechanism are used.

8. Information about HTTPS-based High Availability clustered systems that make up a virtual node can be specified under the Cluster Virtual Node section of the Node Advanced Options window if required. To access this window, click the Advanced Options... button on the Add, Modify, or Copy Node window for a node.

If you have a virtual machine comprised of two or more systems being managed as HTTPS nodes, check the Cluster Virtual Node checkbox and enter the required information for the cluster and its systems.

Enter the cluster HA Resource Group name that identifies the cluster in the mandatory field.

Click the Add button to add the physical systems that make up the cluster package to the Cluster Virtual Node information.

**NOTE**

Only OVO management server features are available for virtual nodes and one agent feature: distribution of policies and instrumentation to the virtual node. Automatically distributes policies and instrumentation to all physical nodes of the virtual node.

The following options cannot be used for virtual nodes:

- Nodeinfo and mgrconf cannot be distributed.
- Agent Sends Alive Packets.
- All software installation and related options.
- Node Type Message Allowed.
- Limit Buffer Size.
The character set for HTTPS nodes is always set to Unicode.

After installing the OVO software on a managed node, you must make sure that the certificates required by HTTPS communication are created and distributed. The default is for these to be generated automatically. These steps are explained in the next section “Creating and Distributing Certificates” on page 154.
Configuring a Windows Installation Server

OVO HTTPS Agents can be fully automatically installed onto Windows systems using an installation server system. An installation server is a regular Windows managed node with an OVO HTTPS agent installed. Once the OVO HTTPS agent is installed, you can install any further Windows HTTPS nodes from the OVO Admin GUI or using inst.sh on the OVO management server without the need to manually execute the opc_inst.vbs utility on the target nodes.

NOTE

It is necessary to set the installation server in the Communication Options window of the target nodes.

The following guidelines describe the specific configurations required for the OVO HTTPS agent acting as installation server:

- The Windows system hosting the OVO agent which acts as installation server must be in the OVO Node Bank and must be of the same communication type (HTTPS) as the target nodes.
- It is recommended to use a dedicated system as an installation server system because it is necessary that the OVO agent acting as the installation server runs with extensive capabilities (see below). This means, that this OVO agent should not receive any policies or instrumentation to avoid accidental or malicious start of functionality with these capabilities.
- The OVO agent must run as a user who is able to access the target systems using standard Windows access mechanisms. In particularly it must be able to copy files to the target system.

To configure an OVO HTTPS managed node to act as a Windows Installation Server, complete the following steps:

1. Install and start a Windows service on the target system. This can be accomplished by making this OVO Agent run as either:
• A domain administrator
• Any other user who has:
  — Networking capabilities.
  — An identical user/password set-up on the target nodes (Windows pass-through authentication).
  — Administrative capabilities on the target nodes.

By default, the OVO HTTPS agents on Windows managed nodes runs as SYSTEM. This means that it is not able to access remote systems. To change the user under which the OVO agent acting as an installation server runs, perform the following steps:

2. Stop the OVO agent with the command:

   ovc -kill

3. Create the Windows user account to be used.

4. Enter the command:

   cscript <InstallDir>\bin\ovswitchuser.vbs -existinguser <user> -existinggroup <group> -passwd <user_pwd>

   This command requires a few minutes to execute and makes the following changes:

   • Change the permissions of OVO data files.
   • Changes the start-up user of the Windows Service.

5. Due to a limitation in ovswitchuser.vbs, complete the following steps:

   a. Open the Control Panel -> Administrative Tools -> Services

   b. Change the Windows User who is configured to run the service HP OpenView Ctrl Service and re-enter the user password.

   c. Confirm that the user has been given the Start as service capability.

6. Start the agent with the command:

   ovc -start
7. Verify that the processes are running and note the user under which they are running as follows:

a. ovc

b. Open the Task Manager and display the user.
Migrating a DCE Agent to an HTTPS Agent

**WARNING**
The major version of your OVO agent software must not be higher than the version of your OVO management server software. For example, an OVO version A.08.00 HTTPS agent cannot communicate with a OVO version A.07.1x management server.

If you are operating in a flexible management environment with A.07.1x and A.08.00 management servers, make sure that all OVO agents remain on version A.07.1x until all management servers have been upgraded to OVO version A.08.00.

**NOTE**
The `opcinfo` file is converted when you upgrade an OVO 7.1 agent to an HTTPS agent. A copy is saved to the local `/tmp/opcinfo.save` file.

To migrate a DCE agent to an HTTPS agent:

1. On the OVO management server:
   
   Prepare the agent profile generation by checking whether the contents of the `inst_defaults_base.ini` file is appropriate for the node. This step should only be necessary once for complete subnets or domains.

2. Select the node from the Node Bank.

   From the menu bar of the OVO Node Bank window Administrator’s UI (see Figure 5-1), open the Modify Node window by selecting:

   Actions: Node -> Modify

   Select the agent type from the Modify Node window. For example, MS Windows (HTTPS).

3. Install the new agent software by selecting:

   Actions: Agents -> Install / Update OVO Software and Configuration
Templates, actions, commands and monitors are only re-installed on the managed node system with the Update OVO Software and Configuration selection.

When you are asked whether the DCE agent should be de-installed, confirm to continue with the HTTPS agent installation.

Local DCE-specific agent configurations are automatically converted to HTTPS agent formats. These include opcinfo settings, ECS data stores and fact stores, Embedded Performance Agent database files.

4. After the agent software installation has completed on the remote node, you can check the status of the installation by entering one of the following commands.

- On the managed node system:
  
  ovc -status

- On the OVO management server system:
  
  opcagt -status <nodename>

A message confirming successful distribution should be displayed in the Message Browser.
Upgrading in a MoM Environment

When upgrading in a MoM environment, there are two main steps to consider:

- Upgrading the OVO management servers to OVO 8.0.
- Upgrading the managed nodes to OVO 8.0 HTTPS agents.

Execute the following steps to upgrade your OVO 7.x MoM environment to an OVO 8.0 MoM environment:

1. Upgrade at least one OVO 7.x management server to OVO 8.0 management server.
2. Migrate the DCE agents to HTTPS agents as described in “Migrating a DCE Agent to an HTTPS Agent” on page 112.

NOTE

The OVO 7.x management server will no longer be able to manage these migrated systems as HTTPS agents are not supported by OVO 7.x management servers.

3. Download the configuration data from the first OVO 8.0 management server using the `opccfgdwn` utility. For more information, refer to To Download the Current OVO A.07.1x Configuration in the HP OpenView Operations Installation Guide for the Management Server.

4. Upload the downloaded configuration data to any additional OVO 8.0 management server using the `opccfgupld` utility. For more information, refer to To Upload the Saved OVO A.07.1x Configuration in the HP OpenView Operations Installation Guide for the Management Server.

5. Set the install flag in the database for your HTTPS agents. Without this, the uploaded nodes cannot be added automatically into the heartbeat polling list, causing problems for heartbeat polling and configuration distribution.

   Enter the command:

   `opcsw -i <https_node_name>`

6. Repeat steps 4 and 5 for all other OVO 8.0 management servers.
7. Establish a trust between two or more managers so that their environments are able to communicate with each other. Complete the steps described in “Certificate Handling for a Second OVO Management Server” on page 59.

8. Create a responsible manager file for HTTPS nodes and deploy it to the agents:

/etc/opt/OV/share/conf/OpC/mgmt_sv/respmgrs/allnodes.bbc

allnodes.bbc has a higher priority than the allnodes file but lower than the <hex_IP_addr> files for an HTTPS node. The file is distributed automatically together with policies, in the same way as the allnodes file with templates.

allnodes.bbc can be empty or contain only a subset of the settings from the allnodes file. An empty allnodes.bbc file means that no MoM configuration is distributed to an HTTPS node and previously deployed MoM configurations are removed if the owner is the same management server that originally distributed the configuration. All OVO management server systems specified in a responsible manager file going to an HTTPS node must use HTTPS as the communication type and have an OvCoreId.
Migrating an HTTPS Agent to a DCE Agent

NOTE

The opcinfo file is converted when you upgrade an OVO 7.1 agent to an HTTPS agent. A copy is saved to the local /tmp/opcinfo.save file.

When migrating an HTTPS agent to a DCE agent, it is not possible to convert the configuration settings into the opcinfo file. You must make a copy of the configuration information from the eaagt namespace. This data can be displayed before removing the HTTPS agent with the command:

`ovconfget`

After installing the DCE agent, manually enter the configuration information into the opcinfo file and delete the = signs between each key and value pair.

To migrate an HTTPS agent to a DCE agent:

1. When migrating an HTTPS agent to a DCE agent, it is not possible to convert the configuration settings into the opcinfo file.

   Display this data before removing the HTTPS agent with the command:

   `ovconfget`

   Make a copy of the configuration information from the eaagt namespace.

2. Select the node from the Node Bank.

   From the menu bar of the OVO Node Bank window Administrator's UI (see Figure 5-1), open the Modify Node window by selecting:

   Actions: Node -> Modify

   Select the agent type from the Modify Node window. For example, MS Windows (HTTPS).

3. Install the new agent software by selecting:

   Actions: Agents -> Install / Update OVO Software and Configuration
Templates, actions, commands and monitors are only re-installed on the managed node system with the Update OVO Software and Configuration selection.

When you are asked whether the HTTPS agent should be de-installed, confirm to continue with the DCE agent installation.

4. After installing the DCE agent, manually enter the configuration information from the HTTPS installation into the opcinfo file and delete the = signs between each key and value pair.

5. After the agent software installation has completed on the remote node, you can check the status of the installation by entering one of the following commands.

- On the managed node system:
  
  ovc -status

- On the OVO management server system:
  
  opcragt -status <nodename>

A message confirming successful distribution should be displayed in the Message Browser.
Installing Agents Manually

In some situations, you may want to install the OVO HTTPS agent software without using the management server. This manual installation enables you to prepare the system to become an OVO managed node when it is later connected to the network. Manual installation is useful if you are preparing many systems in a central location, or if you want to avoid the network connection necessary for standard installation. Manual installation may be necessary for systems behind a firewall or behind an HTTP proxy.

Certificate Installation Tips

If an agent is installed before it is added to the OVO management server node bank, a certificate request is issued from the node, but it remains in the list of pending certificate requests in the Node Certificate Requests window, because it cannot be automatically mapped to any node from the node bank.

It is possible to add a node to the Holding Area from the OVO Node Certificate Requests window by selecting Certificate Request and clicking the Add Node to Node Bank button. The Add Node window opens and you can edit the fields and then add nodes to the Holding Area. Certificate requests are then automatically mapped to that node, but they are not granted. An administrator must manually grant the certificate requests as required.

When a certificate request is granted, the certificate server signs the certificate and sends it to the certificate client. The certificate client now installs the certificate on the node.

NOTE

Remote certificate deployment type can be used during manual agent installation.

After the certificate is installed on the node, either by using remote certificate deployment or by manually importing the certificate to the node, the certificate client notifies the certificate server that the certificate has been successfully installed. The certificate server notifies the certificate server adapter and certificate server adapter then sets the Node Certificate State in the database to Installed.
For more detailed information about handling certificates, refer to “Creating and Distributing Certificates” on page 154.

For troubleshooting certificates handling, refer to “Problems during Certificate Deployment” on page 206.

**To Install an Agent Manually from Package Files**

To install the OVO HTTPS agent on a system that you want to manage as an OVO managed node, follow these steps:

1. **Copy the OVO agent packages, installation script and package description to a temporary directory on the managed node.**

   The files on the OVO management server that you require are:

   - HPOvBbc.<platform>
     HPOvBbc.xml
   - HPOvConf.<platform>
     HPOvConf.xml
   - HPOvCtrl.<platform>
     HPOvCtrl.xml
   - HPOvDepl.<platform>
     HPOvDepl.xml
   - HPOvEaAgt.<platform>
     HPOvEaAgt.xml
   - HPOvPCO.<platform>
     HPOvPCO.xml
   - HPOvPacc.<platform>
     HPOvPacc.xml
   - HPOvPerlA.<platform>
     HPOvPerlA.xml
   - HPOvSecCC.<platform>
     HPOvSecCC.xml
Working with HTTPS Nodes

Configuring HTTPS Nodes

- HPOvSecCo.<platform>
  HPOvSecCo.xml
- HPOvXpl.<platform>
  HPOvXpl.xml
- opc_inst (UNIX) or opc_inst.vbs (Windows)

The following are the optional language packages:

- HPOvLcja.<platform>
  HPOvLcja.xml
- HPOvEaAja.<platform>
  HPOvEaAja.xml
- HPOvEaAes.<platform>
  HPOvEaAes.xml
- HPOvEaAko.<platform>
  HPOvEaAko.xml
- HPOvEaAzS.<platform>
  HPOvEaAzS.xml

The .xml files are common to all architectures.

The depot files for the supported platforms are identified with a platform-specific extension <platform>. The value of <platform> is as follows:

depot.Z Files for HP-UX nodes
sparc.Z Files for Solaris nodes
rpm.gz Files for Linux nodes
msi Files for Windows nodes

The files are located in the following directory on the management server:

/<OvDataDir>/share/databases/OpC/mgd_node/vendor/ \
/vendor/<newarch>/<ostype>/A.08.00.xx/RPC_BBC/

where, for example, <vendor>/<newarch>/<ostype> is:
2. Create a Default Profile.

Create a default profile should with the command:

```
opcswh -create_inst_info <nodenames>
```

For each node from `<nodenames>`, the following file is created:

```
/var/opt/OV/share/tmp/OpC/distrib/<hex_IP_addr>.i
```

The file contains the installation defaults for the node with IP address `<hex_IP_addr>`. The file is automatically copied to the target node via remote agent installation (`inst.sh`) or you can use it for manual agent installation.

After node is added to the OVO database; copy the `<hex_IP_addr>.i` profile file to the managed node system. To activate the profile use one of the following commands:

```
opc_inst -config <hex_IP_addr>.i
```

or

```
opcactivate -config <hex_IP_addr>.i
```

The settings are placed under `local_settings` and have highest priority in the same way as `opcinfo` settings for DCE nodes.

You can maintain these common settings on the OVO management server using the file:

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

A sample configuration file with examples of how to set up parameters is available at:

```
/etc/opt/OV/share/conf/OpC/mgmt_sv/bbc_inst_defaults.sampl
```
Take a copy of the bbc_inst_defaults.sampl, rename it bbc_inst_defaults, and modify in accordance with the syntax specified in the sample file.

3. **Install the Agent.**

Go to the temporary directory to which you have copied the packages and execute the following commands:

For UNIX systems:

a. Change the permissions of the agent installation script to ensure that it can be executed:

```
chmod +x ./opc_inst
```

b. Start the agent installation script by entering:

```
./opc_inst -srv <management_server_name>
```

For Windows systems:

Start the agent installation script by running:

```
opc_inst.vbs -srv <management_server_name>
```

Manually installing the agent software using opc_inst also activates the node. The opc_inst tool installs bits and calls the opcactivate tool. opcactivate sets some initial configuration parameters. A separate activation step is not necessary.

If you want to pre-install the agent on a node system with no immediate configuration and prepare the system for later use, for example, by another department, enter the following command and do not specify an OVO management server:

```
./opc_inst -no_start
```

The agent software is installed but the agent is not started.

When the node needs to be activated and the agent started, enter the command:

```
./opcactivate -srv <srv_name>
```

---

**NOTE**

It is possible to install the OVO agent software after adding the node to an OVO node group.
4. **Examine the logfile for the node:**

   If any errors occurred during installation, correct the problems and reinstall. Errors are written to the native installer logfile for the node. For example on HP-UX, the logfile is at the following location:

   `/var/adm/sw/swagent.log`

   Alternatively, `opc_inst` creates a logfile on all platforms in:

   `/<OvDataDir>/log/install_bbc.log`

5. **On the OVO management server, add the pre-installed nodes to the OVO Node Bank window.**

   Use the following menu sequence:

   Actions-> Node-> Add.

6. **On the OVO management server, add the node to an OVO node group.**

   Drag and drop the node onto a node group in the OVO Node Group Bank window.

   or

   use the `opcnode` tool:

   For example for an HP-UX 11 node, enter the command:

   ```
   /opt/OV/bin/OpC/opcnode -add_node
   mach_type=MACH_BBC_HPUX_PARISC \net_type=NETWORK_IP group_name=<node_group> \node_name=<node_name> node_label=<node_label>
   ```

   Refer to the `opcnode` man page for further details.

7. **Update the database and start heartbeat polling for the node.**

   After the node is connected to the network:

   From the command line, enter the following command on the OVO management server:

   ```
   /opt/OV/bin/OpC/opcsw -installed <node>
   ```

8. **Verify that the OVO agent is running on the managed node.**

   Enter the following:

   ```
   /opt/OV/bin/OpC/opcragt -status <node>
   ```
Working with HTTPS Nodes

Configuring HTTPS Nodes

NOTE

Valid certificates must be installed on the managed node, otherwise the agent will not run and the verification will fail.
Setting Variables in OVO

NOTE

The opcsvinfo file is no longer used by OVO 8.0. It is saved during the upgrade procedure to the directory:

/tmp/save710/

The opcsvinfo file from an OVO 7.1 installation is NOT automatically converted when upgrading to OVO 8.0. If you want to convert the contents of the opcsvinfo files, save the file to a temporary location and use the tool:

/opt/OV/contrib/OpC/opcinfoconv

The opcinfo file is converted when you upgrade an OVO 7.1 agent to an HTTPS agent. A copy is saved to the local /tmp/opcinfo.save file.

To set variables on the OVO management server:

1. Enter the command:

   /opt/OV/bin/ovconfchg -ovrg server -ns opc -set \<var_name>  \<value>

2. Restart server processes.

All relevant variables that were available in the opcsvinfo files are also used by OVO 8.0. The OVO 8.0 schema uses namespaces (the parameter -ns from the example above). All former opcsvinfo variables now have the namespace opc, all former opcinfo/nodeinfo variables on HTTPS nodes have the namespace eaagt. The DCE agents still use the opcinfo files.

You can suffix the namespace by the process name if required. For example, to set the port for the DCE message receiver opcmsgrd, enter the command:

ovconfchg -ovrg server -ns opc.opcmsgrd -set \
OPC_COMM_PORT_RANGE 12345

To read the variables on the OVO management server, enter the command:

/opt/OV/bin/ovconfget -ovrg server \
[ <namespace> [ <var_name> ] ]
which either prints all settings, all settings of a namespace, or one variable.

To read variables on a managed node, use the `ovconfget` command, but without `-ovrg server` option.

To set a variable on an agent use `ovconfchgd` without the `-ovrg server` option.

```
/opt/OV/bin/ovconfget [ <namespace> [ <var_name> ] ]
```

You can delete variables with `ovconfchgd -clear` option.

```
/opt/OV/bin/ovconfget -clear [ <namespace> [ <var_name> ] ]
```

You can find more documentation and examples about configuration settings under:

```
/opt/OV/misc/xpl/conf/defaults/*.ini
```
Installing Agents Using Clone Images

When installing a large number of similar node, it may be advantageous to create a clone image of a typical node configuration and use this as the basis for installing the other nodes.

From an OVO point of view, there are two levels of clones that could be created:

- Agent software installed on OVO managed node system.
- Agent software installed with policies deployed to OVO managed node system.

The clone image should not contain the unique identifier of the original node, the `OvCoreId`. If all cloned nodes contain the same identifier, there will be a significant amount of manual reconfiguration required before these nodes are recognized as individual nodes with no confusion.

To install the OVO agent software using a cloned image, complete the following steps:

1. Install the OVO and configure a node that will be cloned.

2. Remove the `OvCoreID` value of the node to be cloned by executing the following command:

   ```bash
   ovconfchg -ns sec.core -clear CORE_ID
   ```

3. Display all installed certificates from the node to be cloned by executing the following command:

   ```bash
   /opt/OV/bin/ovcert -list
   ```

   The output of the following form is displayed:

   ```
   +----------------------------------------------+
   | Keystore Content                           |
   +----------------------------------------------+
   | Certificates:                               |
   | edb87a09-1511-75ff-13c1-f6aef454aa2b (*)    |
   | edb...                                     |
   +----------------------------------------------+
   | Trusted Certificates:                       |
   | CA_edb66a23-1422-04ff-77c1-f1aef555a1b      |
   | CA_edb...                                  |
   +----------------------------------------------+
   ```
4. Remove all installed certificates from the node to be cloned by executing the following command:

```
/opt/OV/bin/ovcert -remove <certificate name>
```

For example

```
/opt/OV/bin/ovcert -remove \
edb87a09-1511-75ff-13c1-f6aef454aa2b \
CA_edb66a23-1422-04ff-77c1-f1aef555aa1b
```

5. Make a clone image of the system without certificates and the OvCoreID value.

6. Copy the image to the new node.

7. Create a new the OvCoreID value on the new node:

```
ovcoreid -create
```

**NOTE**

Use the -force option if the OvCoreID value was not deleted and it needs to be overwritten.

8. Run the opcactivate command to send a certificate request to the OVO management server:

```
./opcactivate -srv <srv_name>
```

**NOTE**

Care must be taken if policies were already deployed to the node that was cloned. If new nodes created from the clone are configured to report to a different OVO management server than that managing the original node, the policies will no longer be trusted and they are signed by the Certificate Authority of the original OVO management server. To trust these policies, add the hostname of the original OVO management server as a secondary manager in the mgrconf file on the new nodes.
Changing Hostnames and IP Addresses

It is not uncommon for a node to have more than one IP address and hostname. If a node becomes a member of another subnet, you may need to change its IP addresses. In this case, the IP address or fully qualified domain name may change.

In general, on HP-UX and Solaris systems, the IP address and the related hostname are configured in one of the following:

- /etc/hosts
- Domain Name Service (DNS)
- Network Information Service (NIS on HP-UX, NIS+ on Solaris)

If you are moving from a non-name-server environment to a name-server environment (for example, DNS or BIND), make sure the name server can access the new IP address.

Hostnames work within IP networks to identify a managed node. While a node may have many IP addresses, the hostname is used to pinpoint a specific node. The system hostname is the string returned when you use the UNIX `hostname(1)` command.

Manually Changing the Hostname or IP Address of a Managed Node

---

**NOTE**

If you are running OVO in a distributed management server (MoM) server environment, modify the procedure as follows:

- Perform steps 1 to 9 on all management server systems that control or monitor the modified node.
- Perform step 10 all OVO management server systems that refer in any OVO template to the old hostname.

---
NOTE

Service Navigator users must check the service configuration file used in the `opcservice` command. This file may contain hostnames and IP addresses that may need to be changed before using the `opcservice` command again. For more information, see the *HP OpenView Service Navigator Concepts and Configuration Guide*.

To change the hostname or IP address of a managed node, follow these steps:

NOTE

If the IP address change of a node is planned, either the IP address is already known or the node is a DHCP client. Setting the node attribute `System acquires IP dynamically` (DHCP) is a much safer and more convenient route. However, this attribute is only available for HTTPS nodes.

1. Verify that the new IP address and hostname are resolvable on the OVO management server.
2. Verify that the new IP address and hostname are not used by other nodes on the OVO management server.
3. Verify that all OVO management server processes, especially the database processes, are running.

Start the OpenView processes by entering:

```
ove -start

ovstart ovacomm

opcsv -start
```

If the database is not running, start it now by entering:

```
/sbin/init.d/ovoracle start
```
4. Change the IP address or node name of the OVO managed nodes.

On the management server system, for each managed node to be modified, change the IP address or node name of the managed node in the OVO database.

Use one of the following methods:

- **OVO Administrator GUI**
  
  Change the IP address or node name in the Modify Node window of the OVO administrator GUI:

  - **IP Address**
    
    To change the IP address, open the Modify Node window, enter the new IP address in the Hostname field, and press Return. The new IP address is displayed in the IP Address option box.
    
    Click [OK] to save your changes.
  
  - **Node Name**
    
    To change the node name, open the Modify Node window, enter the new node name in the Hostname field, and press Return.
    
    Click [OK] to save your changes.

---

**NOTE**

The node name or IP address must be resolvable on the OVO management server.

---

- **Command Line**

  Change the IP-address/node name using the command line tool opcchgaddr. This is recommended if the node name and IP address are not resolvable on the OVO management server.

  Enter the following:

  ```sh
  /opt/OV/contrib/OpC/opcchgaddr -sync -force \n  -label <label> IP <old_addr> <old_name> IP \n  <new_addr> <new_name>
  ```
-sync  Synchronizes any changes to the hostname or IP address with the OVO runtime components.

-force Name service is not consulted. Database is not checked for duplicate node names.

-label <label> Modifies the label of the node to <label>. The new label is displayed in the Node Bank.

-old_addr  IP address of the old node.

-new_addr  IP address of the new (renamed) node.

-old_name  Name of the old node.

-new_name  Name of the new (renamed) node.

For more information about this command, see the man page opcchgaddr(1M).

5. For IP address changes only on OVO managed nodes (not the management server system), ensure that the new IP address is configured on the managed node.

6. On DCE/NCS nodes only, and for hostname only changes on OVO managed nodes, force OVO to recreate templates from the database by removing cached templates from the last distribution:

   cd /etc/opt/OV/share/conf/OpC/mgmt_sv/templates
   rm -f `find . -type f`

7. On DCE/NCS nodes only, and for hostname only changes on OVO managed nodes, redistribute the templates to all managed nodes as follows:

   a. In one of the main windows, select
      Actions:Agents->Distribute.

   b. In the Install / Update SW & Config... window, select the component [Templates].
c. Select [Force Update] and [Nodes in list requiring update].

d. Select the managed nodes in the Node Bank window, and click [Get Map Selections] in the Install / Update SW & Config... window.

e. Click [OK].

8. Restart the agent on the managed node with the command:

```
/opt/OV/bin/OpC/opcragt -start <node_name>
```


NNM may have already discovered IP and hostname changes. This depends on the NNM configuration and several other timing issues.

On the management server for all OVO managed nodes whose hostname or IP address you want to change, do this:

a. Use the ping command to update OpenView with the changed hostname and IP address:

```
ping <new_name>
```

b. Update the OpenView Topology Database by entering:

```
/opt/OV/bin/nmdemandpoll <new_name>
```

10. Reload the Operator GUI browser.

Reload the OVO administrator’s and operator’s GUI, using the following menu option in any of the main OVO windows:

File: Restart Session

---

**NOTE**

Operators running a Motif GUI and responsible for nodes before and after these nodes have been modified get a popup message but they do not if they are only responsible for the modified node.

Operators running a JAVA GUI and responsible for nodes which have been modified do not receive a message. It is necessary to inform the operators through a reliable channel (for example, an opcmessage).
Automatically Changing the Hostname or IP Address of a Managed Node

This description covers Monitored, Controlled and Message allow nodes as well as nodes with COMMTYPE DCE/NCS or HTTPS, while the descriptions slightly differs in some cases.

To ease this complex process there are some new command line utilities on the OVO Management server.

NOTE

HTTPS Agent software must be installed on the Management Server.

The steps 1 to 9 from the manual procedure above can be performed by the script:

```
/opt/OV/bin/OpC/utils/opc_node_change.pl
```

Sending an opc message is also done by the script and therefore only the browser reload must done manually by the operator.

```
opc_node_change.pl [-h[elp]|-?] \\
-olddate OLD_FQDN -olddaddr OLD_IP_ADDR \\
-neiname NEW_FQDN -newaddr NEW_IP_ADDR[,NEW_IP_ADDR,...] \\
[-nnmupdate -netmask 999.999.999.999 -macaddr \ 
XX:XX:XX:XX:XX:XX \\
[-hook CMDNAME] [-nnmtopofix]]
```

If no NNM update is needed, normally if no NNM functionality is used, update of NNM can be safely ignored. However, if you are not sure, use the -nnmupdate option. Only the basic usage is to pass the node name and IP address known by the OVO management server database as OLD_FQDN (Old Full Qualified Domain Name) and OLD_IP_ADDR and the new values as NEW_FQDN and NEW_IP_ADDR.

If an NNM update is necessary the option -nnmupdate is required. This options needs the information of the netmask and the Adapter/MAC address of the Node! The MAC address can either be passed as option -macaddr in hexadecimal notation with colons or by a callback command line utility passed as parameter of option -hook. The CMDNAME command will get NEW_FQDN and NEW_IP_ADDR as parameters. It must exit with 0 and pass the MAC address by printing MAC=XX:XX:XX:XX:XX:XX to standard out. There is one example hook command in:
Working with HTTPS Nodes
Configuring HTTPS Nodes

/opt/OV/contrib/OpC/opcgetmacaddr.sh, which uses
/opt/OV/bin/snmpget to get the MAC address of the specified node.
This only works with nodes supporting SNMPv2.

The option -nnmtopofix is only needed to fix the NNM configuration.
Use this option, if you encounter problems with nodes which changed
their name or IP address.

**NOTE**
The -nnmtopofix option has a high time and resource consumption.

Comparing Configured Nodes Against Name Resolution

The command line utility opc_chk_node_res.pl compares each
configured node against name resolution. Its location is:

```
/opt/OV/bin/OpC/utils/opc_chk_node_res.pl
```

**NOTE**
Depending on the number of configured nodes and the name resolution
mechanism, the opc_chk_node_res.pl command can result in a high
database and network load.

A mismatch of the configured name or IP address is reported to standard
output and an opcmessage is sent for each incident. The opcmessage
contains the new values for name or IP address if they can be evaluated.
There are some options to limit the number of checks or messages to be
sent.

```
opc_chk_node_res.pl [-h[elp]] [-quiet] [-max ###] \[-check all|managed|external] \[-name FQDN|-addr DOTTED_IP_ADDR]
```

- **-help**  This page.
- **-quiet**  No informational output to STDOUT.
- **-max 200 is def**  Use this option to limit the number of opc
  messages sent by this command.
  Use -1 for unlimited messages.
- **-check all is def**  Use this option to limit checks if desired.
Working with HTTPS Nodes
Configuring HTTPS Nodes

-name FQDN  Use this option to check a single node, specified by the fully qualified domain name.

-addr DOTTED_IP_ADDR  Use this option to check a single node, specified by its IP address (for example, 192.168.1.1).

Parameters of the opcmsg sent can be customized in the script.
Proxies in OVO

Firewall programs and their associated policies, located at a network gateway server, are gateways that are used to protect the resources of a private network from external users. Users of an intranet are usually able to access the approved parts of the Internet while the firewall controls external access to the organization's internal resources.

There are two basic categories of firewalls:

- IP packet filters that work on the network level.
- Proxy servers that work on the application level, for example, a web proxy.

A proxy is a software application that examines the header and contents of Internet data packets and takes necessary action required to protect the systems to which the data is directed. In conjunction with security policies, proxies can remove unacceptable information or completely discard requests.

There are significant security-related advantages of using Application Proxies. These include:

- A fine granularity of security and access control can be achieved as proxies examine packets at the application level. For example, it is possible to restrict specific types of file transfer such as .exe files.
- Proxies can provide protection against “Denial of Service” attacks against the firewall.

There are two commonly cited disadvantages of using proxies:

- Proxies require large amounts of computing resources in the hose system but this is no longer a practical issue as powerful computers are now relatively inexpensive.
- Proxies must be written for specific application programs and there may be programs for which proxies are not easily available.

A proxy server stops and inspects all information before letting it access the internal network. Therefore, by using a proxy, there is no direct connection between an internal network and the “outside” world. Users must authenticate to the proxy to be able to send out information. When a client within the intranet attempts to make a request to the Internet, the proxy actually receives that request. Using Network Address Translation (NAT), the proxy changes the source IP address of the packet.
to that of the proxy server, which hides the identity of the users on the internal network from the outside. If the request meets the requirements of any established policies, the proxy server forwards this request to the desired address. When a response is received, the process is reversed. As long as the incoming request is deemed to be safe, the request is forwarded to the target client on the network. The source address of the response remains unchanged but the destination address is changed back to that of the requesting machine within the firewall. This confers a dramatic increase in security for the network because there is no direct, uncontrolled route to any network systems.

There are two basic types of proxy servers:

- **Single-Homed Host**
  The proxy server has only one network card and address, and it is the responsibility of the Internet router to forward requests to the proxy server and block all other information to the network.

- **Dual-Homed or Multi-Homed Host**
  The proxy server is associated with more than one network card. Requests from the internal network are directed to one of the network cards. Information that comes from the Internet is received by the other network card. There is no routing setup between the network cards, so there is no direct connection between the incoming and outgoing information. The proxy server is responsible for deciding what is sent and to where it is sent.
Configuring Proxies
Most LAN-Internet-LAN architectures can be represented by the following diagram or a subset of the illustration.

Figure 5-4  HTTP Proxy Schematic

Internal LAN-A includes the OVO management server and an HTTP proxy.
A firewall separates the internal LAN from the Internet and the outside world.
A external LAN-B includes HTTPS managed nodes and an HTTP proxy.
The proxy communication can be represented by the following diagram or a subset of the illustration.

**Figure 5-5** HTTP Proxy Infrastructure

A: Direct communication; no Proxy. Firewall must accept all connections from *.internal.mycom.com:* to *.external.mycom.com:TARGET_PORT and all connections from *.external.mycom.com:* to *.internal.mycom.com:SOURCE_PORT.

B: proxy_01 is the proxy in domain internal.mycom.com and can access domain external.mycom.com. **Firewall must accept all connections** from proxy_01.internal.mycom.com:* to *.external.mycom.com:TARGET_PORT.

proxy_02 is the proxy in domain external.mycom.com and can access domain internal.mycom.com. **Firewall must accept all connections** from proxy_01.internal.mycom.com to *.internal.mycom.com:SOURCE_PORT.

C: proxy_01 is the proxy in domain internal.mycom.com. proxy_02 is the proxy in domain external.mycom.com. proxy_01 can access proxy_02 and proxy_02 can access proxy_01. **Firewall must accept all connections** from proxy_01.internal.mycom.com:* to
proxy_02.external.mycom.com:PROXY_B_PORT and proxy_02.external.mycom.com:* to proxy_01.internal.mycom.com:PROXY_A_PORT.

The proxies through which an OVO managed node is to communicate must be specified for each node. This is set in the namespace bbc.http and stored in the bbc.ini file using the ovconfchg command. bbc.ini must not be edited manually.

**Syntax**

```
ovconfchg -ns <namespace> -set <attr> <value>
```

where:

- `-ns <namespace>` Sets a namespace for following options.
- `-set <attr> <value>` Sets an attribute (proxy) and values (port and addresses) in current namespace.

For example:

```
```

Defines which proxy and port to use for a specified hostname.

**Format:**

```
proxy:port +(a)-(b);proxy2:port2+(a)-(b); ...
```

- `a`: list of hostnames separated by a comma or a semicolon, for which this proxy shall be used.
- `b`: list of hostnames separated by a comma or a semicolon, for which the proxy shall *not* be used.

The first matching proxy is chosen.

It is also possible to use IP addresses instead of hostnames so 15.*.*.* or 15:*:*:*:*:*:*:* would be valid as well, but the correct number of dots or colons MUST be specified. IP version 6 support is not currently available but will be available in the future.

```
PROXY=web-proxy:8088-(*.hp.com)+(*.a.hp.com;*)
```

The proxy `web-proxy` is used with port 8088 for every server (*) except hosts that match *.*.hp.com, for example `www.hp.com`. If the hostname matches *.a.hp.com, for example, `merlin.a.hp.com` the proxy server will be used.
Manual Agent Installation Behind a HTTP Proxy

Manual agent installation where the node is behind a proxy must follow the dedicated sequence of steps:

1. Take all necessary files to the system where you want to install the HTTPS Agent software. See “Installing Agents Manually” on page 118 for instructions on manual installation of HTTPS agent software.

2. Start the agent installation script by entering:

   ./opc_inst

   You can also add server and certificate server options to this command.

3. Set the proxy parameters. For example:

   ovconfchg -ns bbc.http -set PROXY
   "web-proxy:8088-(*.mycom.com)+(*.a.mycom.com;*)"

4. When the node needs to be activated and the agent started, enter the command:

   ./opcactivate -srv <srv_name>

Setting Proxies on a Managed Node

To set proxies on an OVO managed node:

1. Manually install the agent software on the managed node system. It will probably not be possible to do a remote installation as the target system cannot yet be reached. See “Installing Agents Manually” on page 118 for instructions on manual installation of HTTPS agent software.

2. Set the proxies over which the OVO agent will communicate with the OVO management server. For example:

   ovconfchg -ns bbc.http -set PROXY
   "web-proxy:8088-(*.mycom.com)+(*.a.mycom.com;*)"

3. Stop all agent processes with the command:

   ovc -kill

4. Restart the agent with the command to register the proxy changes:

   ovc -start
Setting Proxies on the OVO Management Server

To change the proxy settings on the OVO management server:

1. Set the proxies over which the OVO management server will communicate with its managed nodes. For example:
   
   ```
   ovconfchg -ns bbc.http -set PROXY
   "web-proxy:8088-(*.mycom.com)+(*.a.mycom.com;*)"
   ```

2. Stop all OVO processes with the following commands:
   
   ```
   ovstop ovoacomm
   /opt/OV/bin/OpC/ovc -kill
   ```

3. Restart the processes with the following commands to register the proxy changes:
   
   ```
   ovstart ovoacomm
   /opt/OV/bin/OpC/opcsv -start
   /opt/OV/bin/OpC/opcagt -start
   ```
De-installing Agents

You can de-install agents from an HTTPS managed nodes automatically or manually.

De-installing Agents Automatically

To find out how to de-install agents automatically, see the OVO Administrator's Reference.

To De-install an Agent Manually

To de-install an OVO agent from an HTTPS managed node manually, execute the following steps.

For UNIX managed nodes:

1. Go to the installation directory:
   ```
   cd /opt/OV/bin/OpC/install
   ```
2. Enter the following command:
   ```
   ./opc_inst -r
   ```

For Windows managed nodes:

1. Stop all OVO agents running on the managed node.
2. Run the following command:
   ```
   $INSTALLDIR\bin\OpC\install\opc_inst.vbs -r
   ```

De-installation Errors

If errors occur during the de-installation, check the local de-installation log files. Errors are written to the native installer logfile for the node. For example on HP-UX, the logfile is at the following location:

```
/var/adm/sw/swagent.log and /var/adm/sw/swremove.log
```

For Windows managed nodes, the logfile is:

```
%SYSTEMROOT%\temp\inst.log
```

Alternatively, opc_inst creates a logfile on all platforms in:

```
/<OvDataDir>/log/install_bbc.log
```
Virtual Nodes in OVO

Clusters are multiple systems, or nodes, that operate as a unit to provide applications, system resources, and data to users. In modern cluster environments such as Veritas Cluster, Sun Cluster or TruCluster, applications are represented as compounds of resources. Those resources construct a resource group, which represents the application running in cluster environment. Each resource has a special function in this compound.

With OVO 8.0, managing node clusters model has been enhanced. There is now a common mechanism for all OV applications running in cluster environments. OV applications are represented as a collection or group of resources.

Terminology

The following High Availability terms and abbreviations are used in OVO:

Cluster

A group of equivalent systems that are operated under control of a cluster management software such as MC/ServiceGuard (MC/SC), Veritas Cluster, and Sun Cluster.

HA Resource Group

High Availability application name.

Virtual Node

The common name for the group of physical nodes where a given HARG may be operating. As such, it is a subset of the nodes making up a cluster. A virtual node typically has a name and IP address, is known to the name resolution and can be addressed like an ordinary system. A virtual node is part of a cluster and a member of the OVO Node Bank.

Physical Node

This is one single system acting as a potential host for the HARG. A set of physical nodes makes up one virtual node. Each physical node is a member of the OVO Node Bank.
Virtual Node Concepts

A virtual node is a group of physical nodes linked by a common HA Resource Group name. The Cluster Awareness (ClAw) extension of the agents on these physical nodes can switch the policies on a physical node as the package itself switches within the virtual node.

The HA Resource Group name linking the managed node provides the following advantages:

- Events detected in the scope of the HA Resource Group, for example, by policies assigned to the virtual node, may receive that name as the originating node.
- Correct filtering and highlighting on the management station GUI.
- Provide appropriate service names and message key correlations for true management of the cluster.

NOTE

This functionality is only available for HTTPS nodes.

A virtual node can be associated with just one HA resource group name. An HA resource group name can be assigned to more than one virtual node, but these virtual nodes should not share any common physical nodes. This is because any policy assigned to both virtual nodes would receive the same HARG a second time and the cluster awareness of the agent would not be able to distinguish the virtual nodes.

Adding Virtual Nodes to OVO

To add a virtual node, from the OVO Node Bank window:

NOTE

You can enter a node into the node bank as a physical node and later changed into a virtual node by selecting Cluster Virtual Node in the Node Modify window.

A virtual node cannot be directly switched back to a physical node in OVO. To do so, the node must be deleted from the node bank and then added again.
1. Open the Add Node window:
   Actions: Node -> Add

2. Enter the necessary node-related information:
   - Node name
   - IP address
   - Node communication type: HTTPS or DCE
   - Check the Cluster Virtual Node check box
   - Enter a list of physical nodes - no virtual nodes and all nodes must be of the same communication type.
   - HA Resource Group name that the cluster will be hosting

---

NOTE

All nodes that are to be a part of a cluster must also be members of the OVO node bank. They must all share the same node type characteristics (platform, operating system, communication type).

The virtual node must not be a DHCP node.

The physical nodes of a cluster must not be virtual nodes themselves.

---

3. Click [OK] to confirm.

**Configuring Virtual Nodes using opcnode(1m)**

Virtual nodes can also be configured in an OVO node bank by uploading them with the opccfgupld(1m) utility or the opcnode(1m) utility.

The new call parameters added to opcnode(1m):
- `-set_virtual`
- `node_list = "node1 node2 ..."`
- `cluster_package = HARG_name`

Example:
```
./opcnode -set_virtual node_name=ovguest3 node_list="talence ovguest3"
```
Modifying Virtual Nodes in OVO

To modify a virtual node, from the OVO Node Bank window:

1. In the Node Bank window, select the virtual node to be modified.
2. Open the Modify Node window:
   Actions: Node -> Modify
3. Modify the virtual node-related information:
   • Change the HA Resource Group name
   • Change the list of physical nodes

   **NOTE**

   All nodes that are to be a part of a cluster must also be members of the OVO node bank. They must all share the same node type characteristics (platform, operating system, communication type).

   The physical nodes of a cluster must not be virtual nodes themselves.

   4. Click [OK] to confirm.

Deleting Virtual Nodes from OVO

To delete a virtual node from the OVO Node Bank, from the OVO Node Bank window:

1. In the Node Bank window, select the virtual node to be deleted.
2. Delete the selected node:
   Actions: Node -> Delete
Assigning Policies to Virtual Nodes in OVO

NOTE

Policies may be defined to include the user variable `<MSG_GEN_NODE_NAME>`. For policies assigned to an HTTPS virtual node, `<MSG_NODE_NAME>` represents the virtual node name and `<MSG_GEN_NODE_NAME>` the physical node name of the event, if the Custom Message Attributes values for namespace and instance are set.

To assign policies to virtual nodes, from the OVO Node Bank window:

1. In the Node Bank window, select the virtual nodes to which policies are to be assigned from De-assigned/Removed.
2. Open the Assign Templates window:
   Actions: Agents -> Assign Templates
3. Open the Add ... window:
4. Insert the virtual node name and the desired policies.
5. Click [OK] to confirm.

De-assigning Policies from Virtual Nodes in OVO

To de-assign policies from virtual nodes, from the OVO Node Bank window:

1. In the Node Bank window, select the virtual nodes to which policies are to be assigned.
2. Open the Assign Templates window:
   Actions: Agents -> Assign Templates
3. Open the Add ... window:
4. Select the row with the policy/node combination to be removed.
5. Click [OK] to confirm.
Deploying Policies to Virtual Nodes in OVO

Policies assigned to virtual nodes get deployed to the associated physical nodes during an Install & Update Software and Configuration request for the virtual node.

To deploy policies to virtual nodes, from the OVO Node Bank window:

1. In the Node Bank window, select the virtual nodes to which policies are to be deployed.
2. Open the Install & Update Software and Configuration window:
   
   Actions: Agents -> Install & Update Software and Configuration

3. Select the templates to be deployed.
4. Click [OK] to distribute the templates to all physical nodes belonging to the selected virtual node.

Distribution to virtual node automatically includes all associated physical nodes. The related HA Resource Group name is added to all policies that are sent to the managed nodes assigned and belonging to the specified virtual node.

If a physical node is being updated which belongs to other virtual nodes, the HA Resource Group name collection is extended to those nodes. As a result, each policy sent to a physical node has all HA Resource Group names attached for the virtual nodes to which it belongs.

Modifying Policy Configuration on Virtual Nodes in OVO

To modify a policy:

1. Open the policy in the Message Source Templates window.
2. Make the required changes to the policy.
3. Click [OK] to confirm the changes and close the window.

   The changed policy is updated on all physical nodes when a new policy deployment is initiated.
Managing HTTPS Agents on DHCP Client Systems

Dynamic Host Configuration Protocol, or DHCP, enables a DHCP server to dynamically allocate network configurations to computers on an IP network. The primary purpose of this is to reduce the work necessary to administer a large IP network and distributed IP addresses to computers as they are required.

DHCP is a client-server application. When a computer connects to a DHCP server, the server temporarily allocates the computer an IP address. The computer uses this address until the lease expires, at which point it can be replaced with a new IP address.

The main advantage of DHCP is that its addressing scheme is fully dynamic. With a DHCP server running on your network, you can add or move computers around on your network and not have to worry about re-configuring your IP settings.

You can manage OVO HTTPS Agents running on DHCP-Client systems. The OVO solution is not dependent on any specific DHCP or DNS product and is based on the following assumptions:

- Node names must not change. The node name can be used as an identifier of a node, even in a manager-of-manager (MoM) environment.
- DHCP and DNS are synchronized.
- There are a relatively small number of IP address changes per day so no IP Address Change Event (IPCE) Storm strategy is necessary. An OVO Agent sends this event, when it detects an IP address change on one of its network interfaces.
- The Java GUI, and the Administrator and Operator UI processes do not automatically update the IP address changes. Administrators and Operators need to restart their UI processes on receipt of the corresponding warning, to load the latest IP address information.
- DHCP support of agents is configurable for each agent and server.
- Dynamic IP address changes at runtime, not only at startup.

The time between two IP address change checks can be configured by setting the \texttt{IPADDR\_CHECK\_INTERVAL} variable on the managed node.
DHCP Settings in OVO

Variables for DHCP

The following variables are used to configure the DHCP-specific behavior of the management server processes.

OPC_DUMMY_IP_RANGE 1.1.1.*

If the OVO/UNIX management server detects an IP address conflict while processing an IP change request, the next free IP address out of the OPC_IP_DUMMY_IP_RANGE is used. The format of this string is [1-9*].[1-9*].[1-9*].[1-9*]. At least one number must be specified. The default is 1.1.1.*.

OPC_IPCE_RETRY_NUM 10

If none of the IP addresses reported by the node matches those of DNS, the IP address change event is buffered. Each event is processed with a maximum number of retries as specified by the OPC_IPCE_RETRY_NUM variable. The default is 10.

OPC_IPCE_RETRY_INTERVAL 180

After the OPC_IPCE_RETRY_INTERVAL time period has elapsed, all buffered IP change events are processed again. The default is 180 seconds.

opcnode Variables for DHCP

The command opcnode has the following DHCP options:

opcnode -add dynamic_ip=yes|no node_name=<fully qualified domain name>

The option -add includes a parameter dynamic_ip. Setting dynamic_ip to yes configures the OVO management server to accept IP address change events from this new node, in the same way as selecting DHCP in the Node Modify window of the Administrator UI.

opcnode -chg_iptype dynamic_ip=yes|no -node_list=<List of nodes>

Setting dynamic_ip to yes configures the OVO management server to accept IP address change events from this modified node, in the same way as selecting DHCP in the Node Modify dialog of the Administrator UI.
NNM Synchronization Using dhcp_postproc.sh

The dhcp_postproc.sh tool is used by the management server process ovoareqadr after successful processing of an IP address change event. This tool synchronizes NNM after the IP address of a node has been changed. The tool obtains the hostname of the node and its new IP address.

Configuration

Enabling Management of Agents on DHCP Clients

Complete the following steps to enable management of HTTPS agents on DHCP Clients:

1. Ensure that DHCP and DNS are synchronized, for example by updating from the DHCP Server. If synchronization is not achieved, the OVO management server cannot process any IP address change events and it will decrease the overall performance of the system.

2. Configure NNM to process DHCP. This is described in the OVO online help; section entitled Deleting Inaccessible DHCP IP Addresses.

3. Customize /opt/OV/contrib/OpC/dhcp_postproc.sh

Customize the script to suit your environment. The following entries are of particular interest:

```
NETMASK="255.255.248.0"    # netmask
MAXRETRY=5                 # number of retries for opctranm
SLEEP_TIME=10              # sleep this amount of seconds
                      # before the next retry
TRACE="off"               # on=do (or off=do not) create
                      # lots of tracefiles in /tmp
NETMON_TOPO_FIX="OFF"     #off is highly recommended
FORCE_NODEINFO_DIST       #off
```

You may add opcmsg or opcwall calls.
Creating and Distributing Certificates

Certificates are needed for network communication using the Secure Socket Layer (SSL) protocol with encryption. Server and client authentication are enabled. Nodes of the managed environment are identified using certificates. The “SSL handshake” between two nodes only succeeds if the issuing authority of the certificate presented by the incoming node is a trusted authority of the receiving node.

You can install certificates automatically, and manually. Please refer to the following sections for further information.

- “Deploying Certificates Automatically” on page 157
- “Manual Certificate Deployment with Installation Key” on page 168

Certificate installation is monitored with OVO messages. After a certificate request has been granted automatically, a notification message confirming the successful deployment of a certificate is sent to the message browser. If a certificate request is not automatically granted, a message in the message browser indicates the reasons for request denial and the steps that an administrator must take to solve the problem.

Certificates are managed from the Node Certificate Requests window. To open this window, select:

Actions ➔ Node ➔ OVO Certificate Requests

From the Node Certificate Requests window you can:

- Grant, deny, or delete certificate requests.
- Map certificate requests with the corresponding node from the Node Bank.
- Track certificate request flow.
- Add nodes to the Holding Area.
Initiating an action on selected nodes displayed in the node listbox, such as [Grant], [Deny], and [Delete], executes the action and removes the nodes from the listbox. The contents of the list may also be refreshed by the Certificate Server and the window automatically reloads the list every 10 minutes.

Figure 5-6 Node Certificate Requests Window

Node Information in the Node Certificate Requests Window

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hostname</td>
<td>Hostname of the node that initiated the certificate request (not a unique identifier).</td>
</tr>
<tr>
<td>IP Address</td>
<td>IP address of the node that initiated the certificate request (not a unique identifier).</td>
</tr>
<tr>
<td>OvCoreID</td>
<td>The only unique identifier of an OVO HTTPS node. When you grant a request, you also grant all communication originating from the node with this OvCoreID. The hostnames can be changed, but the OvCoreID remains the unique identifier of the node.</td>
</tr>
<tr>
<td>Mapped to</td>
<td>Hostname of the node to which listed certificate requests are mapped. For requests that are not yet mapped, the Mapped to column is empty. Clicking</td>
</tr>
</tbody>
</table>
[Select all Mapped Requests] selects all certificate requests having a hostname listed in the Mapped to column. See “Map to Selected Node” on page 162.

**Platform**  Operating system of OVO managed node.
Deploying Certificates Automatically

The most common certificate deployment method is to let OVO create, grant and distribute certificates automatically. Figure 5-7 illustrates how OVO issues certificates to HTTPS managed nodes.

After the HTTPS agent software is installed on a managed node system, the certificate management client on the node system creates a private key and a certificate request. A secret key is used to encrypt the certificate request which is sent over the network to the server system. Automatic granting is the default configuration and the autogrant interval is set to 30 minutes. If a request arrives after the allowed time interval, it must be handled manually using the Node Certificate Requests Window. If you wish to change this interval, use the following command:

```
ovconfchg -ovrg server -ns opc -set \
OPC_CSA_AUTOGRANT_INTERVAL <time interval in minutes>
```

If the message is encrypted with the correct key, the receiving management server trusts the sender. This does not provide full security, and is not recommended for highly secure environments but is more
secure than transmitting the requests as plain text. This mode is only used for transmitting the certificate request and the signed certificate, which should be a short period of time.

In secure environments, it is recommended that automatic granting of certificate requests is disabled and that an administrator assesses each request before granting those that are valid. You can do this with the command:

```
ovconfchg -ovrg server -ns opc -set \nOPC_CSA_USE_AUTOGRANT <TRUE|FALSE>
```

However, manual installation of certificates is the only fully secure method.

---

**NOTE**

An OpenView secret key is part of the HP Openview HTTPS security software and is used by default for all HP OpenView HTTPS-based applications. Every installation uses the same secret key.

A configurable secret key is a user configured key that replaces the OpenView secret key. This can be done before the management environment is setup. Ensure that every system that may request a certificate is using the same secret key as the certificate server.

Using a configured secret key ensures that a client system is not able to request a certificate from a foreign certificate server system, for example another HP OpenView installation.

---

**NOTE**

The Certificate Server system must be setup and active before certificates can be generated and distributed.

---

To automatically deploy certificates, install the HTTPS agent software on a managed node system. After the installation, the following steps are executed by OVO:

1. A new public/private key pair is generated on the managed node system by the certificate management client.

2. The managed node system initiates a certificate request on the node system.

3. The generated private key is stored in an encrypted file.
4. The certificate request is encrypted with the secret key and sent to the Certificate Server system (using a non-SSL connection as the node system does not yet have a valid certificate).

5. After the certificate request has been decrypted successfully on the Certificate Server it is added to the pool of pending certificate requests and a notification is sent to all registered components, and corresponding entry in the OVO Event Browser is also displayed.

6. The certificate request is either granted or denied by matching certain preconfigured criteria. For example, the request was made within 2 minutes of the HTTPS agent software being installed on the node system.

---

**NOTE**

Granting of a certificate request is the most security sensitive step in this process. The instance that grants the request should have a good reason to do this. An example would be an administrator who is waiting for a request after deploying a package to the node that now requests a certificate from the certificate server.

---

7. If the request is granted, the certificate request is signed by the Certificate Server. The signed certificate is then encrypted with the secret key and sent to the node system.

If the certificate request is denied, the server system sends a message to the node system indicating that the request has been rejected and corresponding entry in the OVO Event Browser is also displayed.

8. The Certificate Client on the node system receives the response. If the request has been granted, it installs the new certificate and is now ready to use SSL for authenticated connections.

If the certificate request has been denied, the Certificate Client stores this information to prevent an automatic retry.
Managing Certificates for HTTPS Managed Nodes

Certificate management is handled from the OVO Certificate Requests window, illustrated in Figure 5-6. To open the OVO Certificate Requests window:

- Click the [Actions] menu in the Node Bank window and select Node: Add..., and then OVO Certificate Requests menu item from any node-related submap.
  
or
- Right-click a certificate-related message and select the OVO Certificate Requests menu item.

Actions Available from the Node Certificate Requests Window

**Grant**
Grant selected certificate request(s). Only mapped requests can be granted. After the operation is completed, the certificate server automatically refreshes the hostname list.

Certificate requests which are not successfully granted remain selected, and an error message is displayed.

If multiple certificate requests are selected, any unmapped requests are ignored and a message is displayed informing you that unmapped certificate requests cannot be granted. If only unmapped certificate requests are selected, the [Grant] button is deactivated (gray).

**Deny**
Deny selected certificate requests. After the operation is completed, the certificate server automatically refreshes the hostname list. You can deny any certificate request, mapped or not. The [Deny] button is active whenever a certificate request is selected.

**Delete**
Delete selected certificate requests. After the operation is completed, the certificate server automatically refreshes the hostname list. You can delete any certificate request. The [Delete] button is active whenever a certificate request is selected.
Select all Mapped Requests
Select mapped certificate requests. This button is always active. If no certificate request from the list of queued requests is selected, pressing this button results in selecting all mapped requests in the list. If one or more certificate requests are selected, pressing this button de-selects all unmapped requests from the originally selected requests.

Select all Unknown Nodes
Select requests originated by nodes that do not exist in the Node Bank. If one or more certificate requests are selected, pressing this button de-selects all nodes that are not in the Node Bank from the originally selected requests.

Add Node to Node Bank
Add nodes from which certificate requests are being originated. This button is active if either of the following conditions are met:

- One or more unmapped certificate requests are selected.
- There are one or more certificate requests where Hostname is not identical to Mapped To, and Hostname cannot be found in the Node Bank.

If only one certificate request is selected, the Add Node window opens with the Hostname already entered and platform type selected.

If more than one certificate requests are selected when this button is pressed, a pop-up confirmation is displayed, warning that multiple nodes will be added to the Node Bank.

Click [OK] and the nodes are added to the Holding Area. After the nodes are added to the Holding Area, a message is sent to the Message Browser. If all nodes are added successfully, a message with severity Normal is sent. If any node is not added successfully, a message with severity Critical and a list of the nodes which failed to be added to the Holding Area is sent.
Click [Cancel] and no node is added to the Holding Area.

Double-clicking a certificate request item opens an Add Node dialog only if one item is selected and the certificate request is unmapped, or Hostname is not identical to Mapped To or cannot be found in the Node Bank.

**Map to Selected Node**

Map selected certificate requests. This button is active only if you select one unmapped certificate request or a request where Hostname is not identical to Mapped To. The system must be an HTTPS OVO node. The successful mapping operation causes the Mapped to hostname to change accordingly.

If you try to map a certificate request to a node with a hostname that is different to the hostname from which the certificate request originated, a pop-up window opens with a warning that you must perform a forced operation.

**OK**

Stop dynamic refresh and close the Node Certificate Request window.
Certificate Generation for Manual Certificate Deployment

Certificates can be deployed totally manually. This avoids sending any certificate-related information over the network before SSL communication is established. The public/private key pair is generated on the certificate server and then transported to the node system. This method is often chosen for highly secure environments where it is undesirable to transmit certificate and key data over a network.

NOTE

The Certificate Server system must be setup and active before certificates can be generated and distributed.

To manually deploy certificates that have been generated on the Certificate Server:

1. If you are dealing with a particularly large environment, you can create the `bbc_inst_defaults` file to maintain common settings for managed node on the OVO management server. The file should be located as follows:

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

   In the namespace `sec.cm.client`, set the deployment type for your nodes to manual by adding an entry of the following type for each node:

   `<IP address> : CERTIFICATE_DEPLOYMENT_TYPE = MANUAL`

   for example:

   `192.168.10.17 : CERTIFICATE_DEPLOYMENT_TYPE = MANUAL`

   The IP address can accept wildcard to specify ranges of nodes.

   For further information, refer to the file:

   `/etc/opt/OV/share/conf/OpC/mgmt_sv/bbc_inst_defaults.sampl`

   See also See “Changing the Default Port” on page 76 and See “Agent Profile” on page 77 for some examples of how to use the `bbc_inst_defaults` file.
2. If installing the OVO HTTPS agent software manually, create a default profile as described in point 2 of “To Install an Agent Manually from Package Files” on page 119.

3. Install the OVO HTTPS agent software on the selected node, using the GUI, manually, or remotely.

4. Make a note of the OvCoreId value assigned to the selected node. OvCoreId can be retrieved by calling one of the following commands:

   - `ovcoreid`
   - `ovconfget sec.core`

When an agent is newly installed using the Administrator’s GUI, a new OvCoreId is created. However, if an OvCoreId is already present in the OVO database for the managed node system, this is used in preference.

When installing the agent software manually, you must create a profile, copy it and the software packages to the managed node system. The profile includes the original OvCoreId from the OVO database. Install the profile with the command:

```
opc_inst -config <profile>
```

**NOTE**

The OvCoreId stored on a remote system can be determined by using the command:

```
bbcutil -ping http://<remote system>
```

provided that the Communication Broker is running on the remote system.

Alternatively, the OvCoreId can be locally displayed with the command:

```
ovcoreid
```

The OvCoreId value stored for the managed node in the OVO database can be displayed with the command:

```
opcnode -list_id node_list=<nodename>
```
5. On the OVO management server system, ensure that the selected node is added to the OVO Node Bank.

6. As an OpenView administrator, create a signed certificate and the corresponding private key for a specific node manually on the Certificate Server system using the opccsacm command line tool. You must provide a password to encrypt the created data.

---

**NOTE**

If certificates must be created before the OVO HTTPS Agent software is installed on the selected node, it is possible to specify the OvCoreId (coreid parameter) in the following command. A OvCoreId is still created and it is stored in the database. The OvCoreId, which is part of the certificate file name, can be retrieved with the command if the managed node is already stored in the OVO database:

```
opcnode -list_id node_list=<node name>
```

This value must then be set on the corresponding node system after the OVO HTTPS Agent software is installed with the command:

```
ovecoreid -set <id> -force
```

If no OvCoreId is already stored, use the value from the managed node:

The OvCoreId stored on a remote system can be determined by using the command:

```
bbcutil -ping http://<remote system>
```

provided that the Communication Broker is running on the remote system.

Alternatively, the OvCoreId can be locally displayed with the command:

```
ovecoreid
```

To create a certificate for the selected node, on the OVO management server system, enter the command:
Creating and Distributing Certificates

The tool asks you to specify a password to encrypt the created certificate. This is later required to decrypt the certificate when importing the certificate to the managed node system.

7. Set the installation type to MANUAL, either in the bbc_inst_defaults file or with the command:

```
ovconfchg -nssec.cm.client -set \nCERTIFICATE_DEPLOYMENT_TYPE MANUAL
```

Copy the file containing the signed certificate, its corresponding private key and the root certificate onto a floppy disk or other portable media.

The default file location directory if the -file option was omitted is:

```
/<OvDataDir>/temp/OpC/certificates
```

The file name takes the following form:

```
<hostname>-OvCoreId.p12
```

8. Go to the node system and stop the agent locally with the command:

```
ovc -stop
```

9. Install the certificate, the trusted root certificates and the private key from the portable media using the ovcert command line tool. Specify the password used in step 5 when requested during installation of the certificate.

To import the certificate, enter the following command:

```
ovcert -importcert -file <file created in step 5>
```

The tool will ask for the password that was provided in step 5.

---

**NOTE**

Access to the medium that contains private keys should be tightly controlled to ensure that only authorized people can use them.

---

10. After installation, delete the certificate installation file from the managed node, and delete the data on the portable medium or store it in a secured place.

11. Start the agent locally with the command:
`ovc -start`

12. Delete the file created for the certificate import from the certificate server system.
Manual Certificate Deployment with Installation Key

Manual certificate deployment with installation key offers the advantage that the private key never leaves the system to which it belongs. However, it requires that some security-related data is transmitted over the network before the certificate can be installed on the node system.

NOTE

The Certificate Server system must be setup and active before certificates can be generated and distributed.

NOTE

When manually generating certificates on the OVO management server and installing certificates with an installation key, you must manually install the OVO agent software on the managed node system. For further information, refer to “Installing Agents Manually” on page 118.

To manually deploy certificates using an installation key:

1. As an OpenView administrator, initiate the creation of a new installation key on the Certificate Server system. Provide a password to encrypt the created key.

   `opccsacm -geninstkey -file <filename> [-pass <password>]`

   The Certificate Server adds the key to its installation key repository and writes it, together with some management information to a file.

2. Copy the file with the installation key information onto a floppy disk or other portable media.

3. Go to the node system and, using the `ovcert` command line tool, initiate a new certificate request. A new public/private key pair is generated. Use the following command:

   `ovcert -certreq -instkey <filename>`

   The encrypted request is sent to the Certificate Server.

   The Certificate Server decrypts the request with the key from its repository. If the correct installation key was used, the Certificate Server automatically grants the request and sends the signed certificate back to the node. Then it removes the installation key from the repository. If an invalid installation key was used, the request is automatically denied.
Working with HTTPS Nodes

Multiple Parallel Configuration Servers

The policies can be used by shared components. Multiple OV products can work with policies on an agent using an owner concept for policies. Multiple parallel configuration servers are supported for HTTPS managed nodes.

Let us now understand when multiple parallel configuration servers are used. A service provider manages the hardware and operating systems of a set of customer systems. The customer himself manages an application on the same set of nodes. Both the service provider and the customer use an OVO management server to manage these systems. The implementation of a solution could be as follows:

- Service provider and customer create their own certificates but agree on a trust, so that the agent accepts action and configuration requests from both OVO management servers.
- Service provider and customer agree on a responsible manager template (mgrconf) file. One party acts as primary manager, the other as a competence center. Both are listed in the authorized managers.
- The competence center is also allowed to deploy configurations. It provides all policies with a specific attribute which can be matched by the message target rules of the responsible manager file. Related messages are then sent to the competence center, all others to the primary manager.

HTTPS managed nodes can support more than one configuration server. This server is denoted as primary manager. You can switch the config server (opcragt -primmgr), which is also known as a backup management server concept. However, if the backup management server is not setup in exactly the same way as the primary management server, the agent may be configured differently when templates and instrumentations are deployed. Instrumentation files from the primary management server remain, if not overwritten by the backup management server. Additional instrumentation files from the backup management server will be deployed and cumulated on the agent. Policies are only replaced, if the primary and backup management server use the same owner string or if policies are identical. All other policies on the agent will remain unchanged, because they belong to different owners. There are two ways to determine if policies are identical or not:
• Check if they have the same policy ID.
• Check if they have the same policy name, policy type and policy version, but different policy ID.

A policy can only be removed by its owner. With regards to policy removal, the following important scenario must be considered:

Let us assume that we have a backup management server scenario. Initially, the primary management server (A) deploys policy (PA) to agent (G). Then policy (PA) has owner (A).

Next, the backup management server (B) deploys the same policy (PA) to the same agent (G). Because the policy is identical, the already installed policy (PA) with owner (A) is removed and re-installed from backup management server B. Now, the reinstalled policy (PA) has owner (B).

Finally, on the primary management server (A), de-assign policy (PA) and issue template distribution to the same agent (G).

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

The OVO competence center concept allows the forwarding of certain types of messages to dedicated servers. However, a competence center is a message concept and cannot manage configuration deployment. backup management servers are defined through the secondary-manager statements in the responsible manager template (mgrconf) file, and competence centers through message target rules and action-allowed-manager entries.

The OpenView responsible manager concept is based on the following terms:

• **OV Access Rights**

  OV components can define access rights. These are the rights to execute actions, deploy files, configure settings. The rights are mapped to pre-configured OpenView roles. It is possible to alter the mappings by changing configuration settings, for example, to stop remote access to a managed node.

• **Assume OV Defined Roles**
A OVO management server can take over an OpenView defined role. The mapping between management servers and roles is defined in the responsible manager policy and in an initial manager setting to allow the responsible manager policy deployment.

- **Local User Role**
  The local user has all rights, assuming appropriate system rights are given, for example `root`.

- **Initial or Authorized Manager Role**
  This manager has all rights and is setup at install time to allow remote access if required. This node is defined by the `MANAGER` and `MANAGER_ID` settings in the security namespace. There can be only one initial manager.

- **Secondary Manager Role**
  A secondary manager has all rights including action execution and configuration deployment. There can be multiple secondary managers defined in the responsible manager policy. The initial manager and the secondary managers make up the group of possible configuration servers.

- **Action-allowed Manager Role**
  An action-allowed manager has no other rights than the action execution right. There can be multiple action-allowed managers defined in the responsible manager policy.

- **Certificate Authority Defined**
  The settings `CERTIFICATE_SERVER` from the security namespace `sec.cm.client` is setup at install time. It defines the system with the certificate authority, which is contacted to get a valid subscribed certificate for the managed node.

### Multiple Configuration Server Setup

Policies which are deployed to HTTPS nodes have an owner attribute:

```
OVO:<server_fully_qualified_name>
```

This means that two OVO management servers can distribute policies to the same agent without creating any problems.
When a backup management server is desired, you can overwrite the default owner string by using the following configuration setting in the opc namespace:

OPC_POLICY_OWNER

Primary- and backup management servers must share the same owner string then. A mixture of backup- and competence center scenarios is possible: backup management servers use the same owner string, competence centers different. However, be aware that only one owner string can be set per management server. If a manager acts as backup for a certain OVO domain, but as competence center for another one, this will not work.

The local policy utility ovpolicy can modify all policies on a system by default. Specify the -owner option to select policies belonging to a specific owner.

To list all policies of any owner, use the command:

ovpolicy -l

To only list the policies for my_srv, use the command:

ovpolicy -l -owner OVO: <my_srv_full_qualified_name>

ovpolicy can also be used to modify owner strings of policies, for example, if the owner string of an OVO management server must be changed without the need to redeploy the configuration for a managed node.

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

Identical policies can be determined by one of the following ways:

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

Identical policies can be modified by multiple servers, independent of the policy owner.

This avoids many instances of the same policy being installed on an agent and avoids multiple messages being created for the same issue.
If multiple servers are used to deploy the same configuration data, they are acting as backup management servers, and their data must be synchronized. Assign one server for configuration development. Download this data and upload it onto the other servers.

To setup a multiple configuration server, refer to the steps in “Environments Hosting Several Certificate Servers” on page 55 to first establish a trust between the multiple configuration servers and then download and upload configurations. For downloading and uploading configurations, refer to the following steps.

Download the node bank configuration on your management server and upload to backup management server as follows:

1. Create a directory hierarchy like the one below:

   ```
   mkdir -p /tmp/<manag_server_name>/C
   ```

2. Create the file `download.dfs` and add the line `NODE_BANK` to that file:

   ```
   vi /tmp/<manag_server_name>/C/download.dfs
   NODE_BANK;
   ```

3. Start the download:

   ```
   opccfgdwn -backup -force download.dfs
   /tmp/<manag_srv_name>
   ```

4. Make a tar archive of downloaded configuration data and transport it via ftp to the backup management server:

   ```
   tar cvf /tmp/<manag_server_name>.tar
   /tmp/<manag_srv_name>
   ```

On the backup management server, import all nodes that are to be switched. A configuration upload also uploads the OvCoreIds for the managed nodes:

1. Untar the tar archive transported to backup management server in previous step:

   ```
   tar xvf /tmp/<managserver_name>.tar
   ```

2. Upload the nodes to the backup management server. Stop the server processes and upload the configuration with the following command:

   ```
   opccfgupld -add -subentity /tmp/<manag_server_name>
3. List all OvCoreIds for uploaded HTTPS agent nodes with the command:

   `opcnode -list_id node_list=<https_agent_node_name>`

4. For all uploaded HTTPS agent nodes, enter the command:

   `opcsw -i <https_agent_node_name>`

5. Add the uploaded nodes to the Node Group, otherwise messages cannot be seen.

If you have a competence center environment, the responsible manager policy, with the owner attribute, can only be installed from one OVO management server. An agent can support only one responsible manager policy.

With regards to the owner concept, the following example show you how the policies are handled between multiple configuration servers.

Let us assume that we have management server A and management server B, policy X and policy Y. Policy X is newly assigned to the agent from both management server A and management server B. Policy Y is newly assigned to the same agent only from management server A.

**Server A and server B use different owner string. Server A use owner string "A". Server B use owner string "B".**

1. Trigger configuration distribution.
   a. From server A, in delta mode and force mode:
      Policy X and Policy Y are deployed and have owner "A".
   b. From server B, in delta mode:
      Nothing is changed for policy X. Since policy X is already installed, it will remain the same and has owner "A". Nothing is changed for policy Y. It still has owner "A".
      From server B, in force mode:
      Policy X is overwritten and has owner "B".
      Nothing is changed for policy Y. It still has owner "A".

2. De-assign policy X and trigger distribution.
   a. If from server A, in delta mode and force mode:
      If policy X has owner "A", it is removed from the agent.
If policy X has owner "B", it remains the same, because of the different owner string.

b. If from server B, in delta mode and force mode:

If policy X has owner "A", it remains the same, because of the different owner string.

If policy X has owner "B", it is removed from the agent.

3. De-assign policy Y from server A, in delta mode and force mode:

Policy Y is removed.

**Server A and server B use same owner string "A".**

1. Trigger config distribution.

   a. From server A, in delta mode and force mode:
      
      Policy X and Policy Y are deployed and has owner "A".
      
      b. From server B, in delta mode:
      
      Nothing is changed for policy X. It still has owner "A". Policy Y is removed.
      
      In force mode:
      
      Policy X is overwritten and still has owner "A". Policy Y is removed.

2. De-assign policy X and trigger distribution.

   a. If from server A, in delta mode and force mode:
      
      Policy X is removed.
      
   b. If from server B, in delta mode and force mode:
      
      Policy X is removed.

3. De-assign policy Y from server A, in delta mode and force mode:

   Policy Y is removed.

The **delta** and **force** distribution modes are also available for multiple server environments. **force** replaces all policies of the calling owner and all identical policies, even though they are owned by different management servers. For non-identical policies, in delta and force mode, a de-assigned policy is only removed by the same owner.
To remove and re-deploy all policies from all OpenView applications from HTTPS nodes, use the command:

`opcragt -distrib -purge -templates <nodename>`

Instrumentation deployment is cumulative. Neither the delta nor the force installation removes any file on the agent, but only updates existing configurations and adds new ones.

If you want a cleanup and re-install all configuration data in the instrumentation directory on the agent, use the command:

`opcragt -distrib -purge -actions -monitors -commands \ <nodenames>`

There is a setting called `OPC_PRIMARY_MGR` in the OVO agent namespace `eaagt`. It is set to the OVO management server hostname with the command:

`opcragt -primmgr`

If `OPC_PRIMARY_MGR` is not set or invalid, the OVO management server denoted by the `MANAGER` setting. Invalid means that the `OPC_PRIMARY_MGR` is not specified as a secondary manager nor as an action-allowed managers, nor is it the initial manager. The `OPC_PRIMARY_MGR` is only a message related setting and it maps to the `$OPC_PRIMARY_MGR` variable which may be used in message target rules of the responsible manager policy so that messages are sent to that OVO management server.

When adding a backup management server to an existing environment managed by OVO, the following two scenarios should be considered.

- You have primary management server A and backup management server B and you switched your management server to backup management server B. Suppose that the primary management server A and backup management server B are synchronized.

  You decided to keep the owner string on the backup management server B, which means that the primary management server A has a different owner string to that of the backup management server B. If management server A and management server B are synchronized, on backup management server B, you can manage all policies, except `mgrconf` (MoM configuration) and `nodeinfo`. `mgrconf` and `nodeinfo` can only be modified or deleted by their owner. Since the primary management server A deployed `mgrconf` and `nodeinfo`, those two policies are owned by management server A. Only the primary
management server A has right to update them. If you need to modify them, manually change the owner attribute in the policy header using the ovpolicy command line tool.

If you are at a management server system and you want to remotely change the owner attribute for mgrconf and nodeinfo on a managed node, execute the following commands:

For nodeinfo:

```bash
ovpolicy -setowner \
OVO:<your_full_qualified_mgmt_server_name> -host \ 
<your.managed_node_name> -poltype configsettings
```

For mgrconf:

```bash
ovpolicy -setowner \
OVO:<your_full_qualified_mgmt_server_name> -host \ 
<your.managed_node_name> -poltype mgrconf
```

If you are at a managed node system and you want to locally change the owner attribute for mgrconf and nodeinfo, you can execute the following command:

For nodeinfo:

```bash
ovpolicy -setowner \
OVO:<your_full_qualified_mgmt_server_name> -poltype \ 
configsettings
```

For mgrconf:

```bash
ovpolicy -setowner \
OVO:<your_full_qualified_mgmt_server_name> -poltype \ 
mgrconf
```

You can verify the changes by executing the following commands:

Remotely:

```bash
ovpolicy -l -host <your.managed_node_name> -owner \
OVO:<your_full_qualified_mgmt_server_name>
```

Locally:

```bash
ovpolicy -l -owner \
OVO:<your_full_qualified_mgmt_server_name>
```
If you want to have a general view of which policy belongs to which management server on your managed node, you can execute the following command:

Remotely:

```
ovpolicy -l -host <your_managed_node_name> -level 2
```

Locally:

```
ovpolicy -l -level 2
```

- If management server A and management server B are not synchronized, different owner string may lead to more policies than expected on your managed node, because a policy can only be removed by its owner. See also “Multiple Parallel Configuration Servers” on page 169. You can remove policies by using the `ovpolicy -remove` command on a managed node.

In this case, using the same owner string on the backup management server has advantages. Not only you can manage all deployed policies, but also it does not lead to unexpected policies remaining on the agent. However, you must still take `mgrconf` and `nodeinfo` policies into account, as described above, unless you deployed them after changing the owner string. The first management server always retains the ownership of these two policies, and only the owner can modify and delete them.

However, we strongly recommend using identical owner strings for both the primary management server and the backup management server. Keep the owner string on primary management server and set primary management server's owner string on backup management server.
Backward Compatibility and the Differences between OVO 7 and OVO 8

Here is a list of changes in the MoM concept and information about backward compatibility:

- Secondary managers have action execution rights on HTTPS agents only. OVO 7.x responsible manager files can be used for OVO 8.0 agents without changes.
- Secondary managers can deploy config data without the primary manager switch on HTTPS agents. On DCE agents you must first call the primary manager switch:
  
  `opcragt -primmgr`

  For message assignment to managers, `opcragt -primmgr` modifies the primary message target manager.

- For configuration deployment, no existing configuration information is removed an HTTPS agent by a new configuration distribution (`opcragt -primmgr` call) from a secondary server. For DCE agents, this is possible if the primary and secondary servers are not identically configured.

- If there are no templates assigned for a node, the server clears all configuration on a DCE agent, but it changes nothing on an HTTPS agent unless it uses the same owner string as the other server. If you are not the primary manager and executed a deploy configuration (not an `opcragt -primmgr` call), nothing happens on a DCE node.

  In mixed agent environments, only one configuration server should be use. This server must execute an `opcragt -primmgr` call before deploying data. In pure HTTPS environments you have more flexibility due to the separation of configuration and message target servers.

- All OVO 7.x MoM templates can also be used on HTTPS agents. However, HTTPS agents cannot communicate with OVO 7.x management servers. Therefore, all OVO management servers that are referenced in the responsible manager policy of an HTTPS agent must be upgraded to OVO 8.0. The MoM configuration file `allnodes.bbc` on the management server ia available to aid migration from OVO 7.x to OVO 8.0. This file has higher priority
than the allnodes file for data deployment to HTTPS nodes. allnodes.bbc should contain only OVO 8.0 management servers. It can be moved to allnodes, when all servers are updated.

- All OVO management servers which are referenced in a responsible managers policy must be added to the node bank, and their core ID must be added to the database. OvCoreIds are automatically added to the responsible managers policy during deployment. The authorization of servers on HTTPS managed nodes is bases on authorized OvCoreIds.

- If you setup a MoM environment with HTTPS nodes, some certificate related configurations must be made. For details see “Environments Hosting Several Certificate Servers” on page 55.
A Troubleshooting HTTPS-based Communication
Troubleshooting

If communication between OVO/UNIX Management Server and an HTTPS Agent appears to be interrupted, for example, messages do not arrive at the Message Browser, or software or instrumentation is not distributed, execute the appropriate troubleshooting steps as described in the following sections.

Before you continue with the described actions, you should be familiar with the new HTTPS agent and the underlying communication concepts such as certificates.

This guideline describes possible actions to identify and solve HTTPS communication problems between OVO management servers, Certificate Authority Servers and OVO managed node agents.

It is assumed, that the OVO HTTPS agent software is installed, but there is a problem in the communication between OVO managed nodes and OVO management servers in one or both directions.

In most installations, the OVO management server and Certificate Authority servers are installed on the same system.

Troubleshooting problems encountered with the communication between a OVO 8.0 management server and an OVO HTTPS agents is split into the following areas:

- Troubleshooting Tools
- Logging
- Troubleshooting Processes

Troubleshooting Tools

Ping an HTTPS-Based Application

HTTPS-based applications can be pinged to test if the application is active and responding. A ping may be executed against an application whether or not it has SSL enabled.

The `bbcutil` utility supports a `-ping` command line parameter that can be used to ping an HP OpenView HTTPS-based application.
Use the following command to ping a specified HTTPS-based application:

\[<OvInstallDir>/bin/bbcutil -ping [<hostname_or_ip_addr>] [count]\]

For example:

HTTP  \[bbcutil -ping http://...\]

HTTPS  \[bbcutil -ping https://...\]

Checks whether the communication service on the node specified by
\(<hostname_or_ip_addr>\) is alive. If the hostname or IP address is
omitted, localhost is assumed. An optional loop count can be specified
after the hostname or IP address which causes the ping command to be
repeated by the number of times specified.

See the \(bbcutil\) man page for details of the command line parameters.

**Display the Current Status of an HTTPS-Based Application**

An HTTPS-based application at a specified location can be requested to
display its current status.

Use the following command to query a specified application:

\[bbcutil -status <hostname_or_ip_addr:port>\]

Queries the communication server located at the hostname:port specified
by \(<hostname_or_ip_addr:port>\) for details about the current state of
the server.

See the \(bbcutil\) man page for details of the command line parameters. If
a port is not specified, the port number of the Communication Broker is
used.

**Display All Applications Registered to a Communication Broker**

The Communication Broker at a specified location can be requested to
display all applications that are registered to it.

Use the following command to list all applications that are registered to
the specified Communication Broker:

\[bbcutil -registrations|-reg <hostname_or_ip_addr>\]

Queries a Communication Broker on the node specified by
\(<hostname_or_ip_addr>\) and displays a list of all registered
applications. If the hostname or IP is omitted, localhost is assumed.
Troubleshooting HTTPS-based Communication

Troubleshooting

See the `bbcutil` man page for details of the Communication Broker command line parameters.

**What String**

All executables contain a detailed UNIX-style `what` string that can be used to determine the precise version of the HTTPS-based communication software installed. Microsoft Windows executables also contain standard property strings.

**List All Installed OV Filesets on an HTTPS Node**

The `ovdeploy` tool can be used to list the installed OpenView products and components. The following three levels of information can be displayed:

- Basic inventory
- Detailed inventory
- Native inventory

The following sections illustrate how to list the inventory and show examples of the output.

**Basic Inventory**

To display basic inventory information, enter the following command:

```
ovdeploy -inv -host <hostname>
```

For example:

```
ovdeploy -inv -host hp_System_002
```

<table>
<thead>
<tr>
<th>NAME</th>
<th>VERSION</th>
<th>TYPE</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARCHITECTURE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>HP OpenView HTTP Communication</td>
<td>05.00.070</td>
<td>package</td>
</tr>
<tr>
<td>Windows 4.0 5.0 5.1 5.2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>HP OpenView Deployment</td>
<td>02.00.070</td>
<td>package</td>
</tr>
<tr>
<td>Windows 4.0 5.0 5.1 5.2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>HP OpenView Security Certificate Management</td>
<td>01.00.070</td>
<td>package</td>
</tr>
<tr>
<td>Windows 4.0 5.0 5.1 5.2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>HP OpenView Security Core</td>
<td>02.00.070</td>
<td>package</td>
</tr>
<tr>
<td>Windows 4.0 5.0 5.1 5.2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Detailed Inventory

To display detailed inventory information, enter the following command:

```
ovdeploy -inv -all -host <hostname>
```

For example:

```
ovdeploy -inv -all -host hp_System_002
```

```
<?xml version='1.0' encoding='UTF-8' standalone='yes'?>
<inventory
 xmlns="http://openview.hp.com/xmlns/depl/2003/inventory">
<host>hpspi002.bbn.hp.com</host>
<date>Thursday, October 30, 2003 12:24:48 PM</date>
<package>
  <name>HP OpenView HTTP Communication</name>
  <version>05.00.070</version>
  <systemtype>IA32</systemtype>
  <ostype>Windows</ostype>
  <osvendor>MS</osvendor>
  <osversion>4.0 5.0 5.1 5.2</osversion>
  <osbits>32</osbits>
  <nativeinstallertype>msi</nativeinstallertype>
</package>
<package>
  <name>HP OpenView Deployment</name>
  <version>02.00.070</version>
  <systemtype>IA32</systemtype>
</package>
...
```

Native Inventory

To display native inventory information, enter the following command:

```
ovdeploy -inv -it native -host <hostname>
```

For example:

```
ovdeploy -inv -it native -host hp_System_002
```

<table>
<thead>
<tr>
<th>NAME</th>
<th>VERSION</th>
</tr>
</thead>
<tbody>
<tr>
<td>WebFolders XP</td>
<td>9.50.5318</td>
</tr>
<tr>
<td>HP OpenView Core Library</td>
<td>2.50.70</td>
</tr>
<tr>
<td>HP OpenView Certificate Management Client</td>
<td>1.0.70</td>
</tr>
<tr>
<td>HP OpenView HTTP Communication</td>
<td>5.0.70</td>
</tr>
<tr>
<td>ActivePerl 5.6.1 Build 633</td>
<td>5.6.633</td>
</tr>
<tr>
<td>HP OpenView Deployment</td>
<td>2.0.70</td>
</tr>
<tr>
<td>Microsoft FrontPage Client - English</td>
<td>7.00.9209</td>
</tr>
</tbody>
</table>
Standard TCP/IP Tools

If SSL is not enabled, standard TCP/IP tools such as telnet can be used to contact HP OpenView HTTPS-based application. To use telnet to ping an HTTPS-based application execute the following commands:

To use telnet to ping an HTTPS-based application, execute the following commands:

```
telnet <host> <port>
PING /Hewlett-Packard/OpenView/BBC/ping HTTP/1.1
```

Two carriage returns are required after the PING input line to telnet.

To end the telnet session, enter `control-D` and `Return`:

```
telnet <host> <port>
PING /Hewlett-Packard/OpenView/BBC/ping HTTP/1.1
```

The output takes the following form:

```
HTTP/1.1 200 OK
content-length: 0
content-type: text/html
date: Thu, 08 Aug 2002 08:20:24 GMT
senderid: fd7dc9c4-4626-74ff-9e5a09bffbae
server: BBC X.05.00.01.00; ovbbccb 05.00.100
```

HTTP status 200 OK indicates the HTTPS-based application has recognized the request and successfully responded. Other status may indicate a failure in the request or other error.

For a list of error codes, refer to:

```
http://www.w3.org/Protocols/rfc2616/rfc2616.html
```

RPC Calls Take Too Long

If an RPC call takes longer than the default timeout of 5 minutes, the following error messages may be displayed, for example for policy installation:

```
ERROR:   General I/O exception while connecting to host '<hostname>'.
         (xpl-117) Timeout occurred while waiting for data.
```

```
ERROR:   The Configuration server is not running on host '<hostname>'.
         Check if the Configuration server is in state running.
         (bbc-71) There is no server process active for address:
         https://<hostname>/com.hp.ov.conf.core/bbcrpcserver
```
This may happen if 1000 policies are installed using the PolicyPackage interface from OvConf or if the connection or target-machine is slow.

To prevent this the communication timeout (response timeout) can be changed.

On the target system, enter the following command with the required time out value:

`ovconfchg -ns bbc.cb -set RESPONSE_TIMEOUT <seconds>`

On the OVO management server, enter the following command with the required time out value:

`ovconfchg -ns bbc.http.ext.conf -set RESPONSE_TIMEOUT <seconds>`

---

**NOTE**

The `RESPONSE_TIMEOUT` parameter must be set on both nodes.

---

**Logging**

Errors in violation of security rules are recorded in a logfile. For HTTPS-based servers, all client access can be additionally logged, if enabled.

To enable logging of all client access, set the following parameter value in the log file `/var/opt/OV/log/System.bin` using the command:

`ovconfchg -ns bbc.cb -set LOG_SERVER_ACCESS true`

This will log all access to the Communication Broker. To view the logs, open the text file:

`<OvDataDir>/log/System.txt`

You can additionally log access to all OV Communication Broker servers using the command:

`ovconfchg -ns bbc.http -set LOG_SERVER_ACCESS true`

You can additionally log all client access to the the configuration and deployment application using the command:

`ovconfchg -ns bbc.http.ext.conf -set LOG_SERVER_ACCESS true`
Communication Problems between Management Server and HTTPS Agents

The most likely areas where communication problems may be experienced are divided into the following sections:

- “Basic Network Troubleshooting” on page 188
- “Basic HTTP Communication Troubleshooting” on page 190
- “Troubleshooting Authentications and Certificates in HTTP Communication” on page 197
- “Troubleshooting OVO Communication” on page 202

Basic Network Troubleshooting

Basic network troubleshooting uses the following commands:

```bash
ping <SYSTEMPATH>/ping
nslookup <SYSTEMPATH>/nslookup
telnet <SYSTEMPATH>/telnet
ovgethostbyname <INSTALLDIR>/bin/ovgethostbyname
```

(for use on Solaris systems only in place of nslookup)

NOTE

The actions described below may not work if communication between an OVO management server or Certificate Authority server and OVO managed node has to pass:

- Firewalls
- NATs
- HTTP Proxies

Contact your Network Administrator for more information.
To check for basic network problems, complete the following steps:

1. Check if the name resolution for the OVO management server, Certificate Authority server and OVO managed node is consistent on all affected systems.

   Use ping, and nslookup (on Solaris: ovgethostbyname) with the Fully Qualified Domain Name (FQDN) on all systems as targets.

   \texttt{bbcutil -gettarget <nodename>}

2. Check if all systems (OVO management server, Certificate Authority server and OVO managed node) are accessible.

   Use one of the following commands:
   
   \begin{itemize}
     \item \texttt{<OvInstallDir>/bin/bbcutil -ping <FQDN>}
     \item \texttt{telnet <FQDN>}
   \end{itemize}

3. Check if HTTP communication is working by using a Web browser to connect to the Communication Broker. The Communication Broker, \texttt{ovbbcch}, must be running for this check.

   To retrieve the assigned \texttt{<AGENT-BBC-PORT>} value, enter the command:

   \texttt{bbcutil -getcbport <agenthostname>}

   For example, if you enter the command:

   \texttt{bbcutil -getcbport mysystem.mycom.com}

   Output of the following form is displayed:

   \texttt{mysystem.mycom.com:8008}

   On the OVO management server system, open a Web browser and enter the following URL:

   \begin{verbatim}
   http://<OVO managed node>:<AGENT-BBC-PORT>)/
   Hewlett-Packard/OpenView/BBC/
   \end{verbatim}

   The default port number for \texttt{<AGENT-BBC-PORT>} is 383.

   Repeat this step from the managed node to the OVO management server:

   \begin{verbatim}
   http://<OVO management server>:<AGENT-BBC-PORT>)/
   Hewlett-Packard/OpenView/BBC/
   \end{verbatim}
The HP OpenView BBC Information Modules page should appear and allow you to check ping and status or list registered services and OV resource groups (ovrg).

Basic HTTP Communication Troubleshooting

Basic HTTP communication troubleshooting uses the following commands:

- `ovc` &lt;INSTALLDIR&gt;/bin/ovc
- `ovconfget` &lt;INSTALLDIR&gt;/bin/ovconfget
- `ovbbccb` &lt;INSTALLDIR&gt;/bin/ovbbccb
- `ps` &lt;SYSTEMPATH&gt;/ps

**NOTE**

Even if the communication between OVO management server or Certificate Authority server and OVO managed node has to pass:

- Firewalls
- NATs
- HTTP Proxies

...the following actions must work! If they do not, contact your Network Administrator for more information.

**NOTE**

If the communication between OVO management server or Certificate Authority server and OVO managed node is not allowed to pass through the firewalls, one or more HTTP Proxies must be used (see the corresponding sections).
To check for HTTP communication problems, complete the following steps:

1. On all systems, the OVO management server, Certificate Authority server and OVO managed node, check if:

   The OV Communication Broker `ovbbccb` is running with the following commands:

   `ovc -status`

   The `ovbbccb` process must be listed as running. The output takes the following form:

<table>
<thead>
<tr>
<th>Process Name</th>
<th>Type</th>
<th>PID</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>ovbbccb</td>
<td>CORE</td>
<td>2786</td>
<td>Running</td>
</tr>
<tr>
<td>ovconfd</td>
<td>CORE</td>
<td>2787</td>
<td>Running</td>
</tr>
<tr>
<td>ovcs</td>
<td>SERVER</td>
<td>3024</td>
<td>Running</td>
</tr>
<tr>
<td>coda</td>
<td>AGENT</td>
<td>2798</td>
<td>Running</td>
</tr>
<tr>
<td>opcmsga</td>
<td>AGENT,EA</td>
<td>2799</td>
<td>Running</td>
</tr>
<tr>
<td>opcacta</td>
<td>AGENT,EA</td>
<td>2800</td>
<td>Running</td>
</tr>
<tr>
<td>opcmsgi</td>
<td>AGENT,EA</td>
<td>2801</td>
<td>Running</td>
</tr>
<tr>
<td>opcle</td>
<td>AGENT,EA</td>
<td>2805</td>
<td>Running</td>
</tr>
<tr>
<td>opcmona</td>
<td>AGENT,EA</td>
<td>2806</td>
<td>Running</td>
</tr>
<tr>
<td>opctrap</td>
<td>AGENT,EA</td>
<td>2810</td>
<td>Running</td>
</tr>
</tbody>
</table>

   `ps <OPT> | grep ovbbccb`

   `ovbbccb` must be listed.

   `<OvInstallDir>/bin/bbcutil -status`

   Status of `ovbbccb` must be ok.
NOTE

Make a note of the ports listed using the command:

```
bbcutil -getcbport <hostname>
```

- on OVO managed node as `<AGENT-PORT>`
- on OVO management server as `<MGMT-SRV-PORT>`
- on Certificate Authority server as `<CA-SRV-PORT>`

You can start the Communication Broker with the command:

```
ovc -start
```

No error messages should be displayed.

If the `ovbccc` process is not running:

a. Check the logfile for error messages in the file:

   ```
   <OvDataDir>/log/System.txt
   ```

b. Start the Communication Broker with the command:

   ```
   <OvInstallDir>/bin/bbcutil -nodaemon -verbose
   ```
   
   If there is any problem, errors are displayed in detail at startup.
   The port number it uses is also displayed on startup.

c. For more detailed output use the command:

   ```
   OVBBC_TRACE=true <OvInstallDir>/bin/ \\
   bbcutil -nodaemon -verbose
   ```
   
   This displays a very significant amount of detailed information.
   This detail can also be obtained using OV tracing.

2. Check the configuration of the Communication Broker port settings
   with the following commands:

   a. Lists all Communication Broker ports:

      ```
      bbcutil -getcbport <hostname>
      ```

   b. Check if the default `DOMAIN` parameter is correctly set for the
      nodes using the command:

      ```
      ovconfget bbc.http DOMAIN
      ```
This should be set to the default domain, for example, myco.com. This parameter may be used to find a match for the parameters configured in step 2.a above.

c. Check if a process has the Communication Broker port open and is listening for connections using the command:

```
netstat -an | grep \.383
```

You should see something similar to (varies on each platform):

```
tcp 0 0 *\.* 383 *\.* LISTEN
```

LISTEN verifies that a process is listening on the specified port. If this is displayed and the Communication Broker is not running, another process is using the port and the Communication Broker will not startup. This can be verified with steps 1.a and 1.b.

3. Check the HTTP Communication capabilities by entering the following commands.

On the OVO management server and the Certificate Authority server:

```
<OvInstallDir>/bin/bbcutil -ping http://OVO managed node:<AGENT-PORT>/
```

On the OVO managed node:

```
<OvInstallDir>/bin/bbcutil -ping \ http://OVO management server:<MGMT-SRV-PORT>/
<OvInstallDir>/bin/bbcutil -ping \ http://Certificate Authority server:<CA-SRV-PORT>/
```

Each call should report:

```
status=eServiceOK
```

4. Check if the nodes have the correct Communication Broker port configuration. Do not specify a port number in the URI. OV communication must be able to resolve the Communication Broker port number on its own. If the ping works with the port number, but does not work without the port number, the local node is not correctly configured. Go back to step 2.

5. Check if the HTTP Proxy is correctly configured using the command:

```
bbcutil -gettarget <nodename>
```

For example, if you enter the command:
Troubleshooting HTTPS-based Communication

Troubleshooting

```
bbcutil -gettarget mysystem.mycom.com
```

Output of the following form is displayed:

```
Node: mysystem.mycom.com:8008 (14.133.123.10)
```

If a proxy is configured, it will be displayed.

For example, if you enter the command:

```
bbcutil -gettarget www.mycom.com
```

Output of the following form is displayed:

```
HTTP Proxy: web-proxy:8008 (14.193.1.10)
```

```
ovconfget bbc.http PROXY
```

Although not recommended, applications may set their own private PROXY setting. The above setting is valid for the whole node. An individual application may override this value in its own private namespace:

```
ovconfget bbc.http.ext.<comp id>.<appname>
```

If the `<comp id>` or `<appname>` is not known, check using ovconfget the entire configuration for all proxy settings in the namespaces starting with:

```
bbc.http.ext
```

6. Check on the OVO management server and the Certificate Authority server systems that the proxy is working and supports the CONNECT command.

```
NOTE
```

The blank lines are important.

On some platforms, it may not be possible to echo commands typed into telnet.

```
Enter the command:
```

```
telnet <proxy> <proxy port>
CONNECT <AGENT>:<AGENT PORT> HTTP/1.0
PING /Hewlett-Packard/OpenView/BBC/ HTTP/1.0
```
To exit telnet, enter **Control-D**

The output should be similar to the following. If the Communication Broker is up and running on the target node, the HTTP status should be **200 OK**.

```
HTTP/1.1 200 OK
cache-control: no-cache
content-type: text/html
date: Fri, 06 Feb 2004 15:15:02 GMT
senderid: fd7dc9e4-4626-74ff-084a-9e5a09bffbae
server: BBC 05.00.101; ovbbccb 05.00.101HP OpenView BBC
```

Information Modules:

```
Node: ping.bbn.hp.com
Application: ovbbccb
Version: 05.00.101
Modules: ping
status
services
ovrg
```

Connection closed by foreign host.

7. Check on the OVO managed node that the proxy is working and supports the CONNECT command.

**NOTE**

The blank lines are required.

On some platforms, it may not be possible to echo commands typed into telnet.

Enter the command:

```
telnet <proxy> <proxy port>
CONNECT <MGMT-SRV>:<MGMT-SRV PORT> HTTP/1.0
PING /Hewlett-Packard/OpenView/BBC/ HTTP/1.0
```

or

```
telnet <proxy> <proxy port>
CONNECT <CA-SRV>:<CA-SRV PORT> HTTP/1.0
PING /Hewlett-Packard/OpenView/BBC/ HTTP/1.0
```
8. Enable logging for HTTP access to the Communication Broker.

```
ovconfchg -ns bbc.cb -set LOG_SERVER_ACCESS true
```

This will log all access to the Communication Broker. To see the logs use:

```
ovlogdump <OvDataDir>/log/System.bin
```

You can additionally log access to all OV servers using:

```
ovconfchg -ns bbc.http -set LOG_SERVER_ACCESS true
```
Troubleshooting Authentications and Certificates in HTTP Communication

Troubleshooting Basic HTTP communication uses the following commands:

```
ovc        <INSTALLDIR>/bin/ovc
ovconfget  <INSTALLDIR>/bin/ovconfget
ovconfchg  <INSTALLDIR>/bin/ovconfchg
ovcoreid   <INSTALLDIR>/bin/ovcoreid
ovcert     <INSTALLDIR>/bin/ovcert
bbcutil    <INSTALLDIR>/bin/bbcutil
```

To check for authorization and certificate related HTTP communication problems, complete the following steps:

1. Check the OvCoreID of each system.
   
   On the OVO management server or the Certificate Authority server, enter the command:
   
   `ovcoreid -ovreg server`
   
   On OVO managed nodes, enter the command
   
   `ovcoreid`
   
   Make a note of each of the displayed OvCoreID values:
   
   - `<MGMT-SRV-COREID>`
   - `<CA-SRV-COREID>`
   - `<AGENT-COREID>`

2. Check the certificates on the OVO management server or Certificate Authority server and on OVO managed nodes using the following command:

   `ovcert -list`
There are 3 certificates on the OVO management server system or Certificate Authority system:

- OVO management server certificate
- Certificate authority certificate
- Node/agent certificate

When an OVO management server is installed on a cluster (high availability environment), the certificates of the OVO management server and the agent on the management server are not the same. On non-cluster installations, the certificates must be identical.

On each system there must be at least following Certificates.

On OVO managed nodes:

| Certificates: |
| <AGENT-COREID> (*) |

On the OVO management server or the Certificate Authority server:

| Certificates: |
| <MGMT-SRV-COREID> | <CA-SRV-COREID> (*) |

On all systems:

| Trusted Certificates: |
| <CA-SRV-COREID> |

The (*) signifies that the private key for the certificate is available.

If one of the certificates is missing, refer to “Creating and Distributing Certificates” on page 154 and generate the required certificates.
Troubleshooting HTTPS-based Communication

To get more detailed info about the installed certificates, use the following commands:

On OVO managed node:

```
ovcert -check
```

On the OVO management server:

```
ovcert -check -ovrg server
```

An example of the output is shown below:

```
OvCoreId set                       : OK
Private key installed              : OK
Certificate installed              : OK
Certificate valid                  : OK
Trusted certificates installed     : OK
Check succeeded.
```

To check that the installed certificates are valid, use the following command and make sure that the current date is between the valid from and valid to dates of the installed certificates:

```
ovcert -certinfo <CertificateID>
```

**NOTE**

The CertificateID of a trusted certificates is the OvCoreID of the certificate server prefixed with a CA_.

An example of the output is shown below:

```
# ovcert -certinfo 071ba862-3e0d-74ff-0be4-b6e57d0058f2
Type       : X509Certificate
Subject CN : 071ba862-3e0d-74ff-0be4-b6e57d0058f2
Subject DN : L: alien2.ext.bbn.com
            O: Hewlett-Packard
            OU: OpenView
            CN: 071ba862-3e0d-74ff-0be4-b6e57d0058f2
Issuer CN  : CA_99300c4e-f399-74fd-0b3d-8938de9900e4
Issuer DN  : L: tccbm054.bbn.hp.com
            O: Hewlett-Packard
            OU: OpenView
            CN: CA_99300c4e-f399-74fd-0b3d-8938de9900e4
Serial no.  : 04
Valid from  : 01/27/04 12:32:48 GMT
Valid to    : 01/22/24 14:32:48 GMT
```
An HTTPS agent is also installed on an OVO management server system. If calling `ovcert -list` on an OVO management server system, you are given the certificate details of the agent on the OVO management server system.

3. Check the HTTPS communication capabilities using the following commands.

The following actions must work even if communication between an OVO management server or a Certificate Authority server and an OVO managed node has to pass:

- Firewalls
- NATs
- HTTP Proxies

If they do not, contact your Network Administrator for more information.

If the communication between OVO management server or Certificate Authority server and OVO managed node is not allowed to pass through the firewalls, one or more HTTP Proxies must be used (see the corresponding sections).
Troubleshooting HTTPS-based Communication

Troubleshooting

On an OVO management server or Certificate Authority server:

<OvInstallDir>/bin/bbcutil -ping \
https://<OVO managed node name>:<AGENT-PORT>/

On an OVO managed node:

<OvInstallDir>/bin/bbcutil -ping \
https://<OVO management server name>:<MGMT-SRV-PORT>/

<OvInstallDir>/bin/bbcutil -ping \
https://Certificate Authority server:<CA-SRV-PORT>/

Each call should report:

status=eServiceOK

The reported OvCoreID must match with the OvCoreIDs that you noted in the first step:

coreID=<COREID>
Troubleshooting HTTPS-based Communication

Troubleshooting

Troubleshooting OVO Communication

Troubleshooting OVO communication uses the following commands:

- `ovc` : `<INSTALLDIR>/bin/ovc`
- `ovconfget` : `<INSTALLDIR>/bin/ovconfget`
- `ovconfchg` : `<INSTALLDIR>/bin/ovconfchg`
- `ovcoreid` : `<INSTALLDIR>/bin/ovcoreid`
- `ovpolicy` : `<INSTALLDIR>/bin/ovpolicy`
- `ovcs` : `<INSTALLDIR>/bin/ovcs`
- `opcagt` : `<INSTALLDIR>/bin/OpC/opcagt`
- `opcragt` : `<INSTALLDIR>/bin/OpC/opcragt`
- `opcsccsa` : `<INSTALLDIR>/bin/OpC/opcsccsa`
- `opcsam` : `<INSTALLDIR>/bin/OpC/opcsam`
- `opcsv` : `<INSTALLDIR>/bin/OpC/opcsv`
- `opcnnode` : `<INSTALLDIR>/bin/OpC/opcnnode`
- `opc` : `/usr/bin/OpC/opc`

To check for OVO communication problems, complete the following steps:

1. OVO managed nodes must be in the OVO Node Bank.
2. The Fully Qualified Domain Name (FQDN) of the OVO managed node must match.
3. The communication type of the OVO managed node must be HTTPS.
4. The OvCoreID of the OVO managed node must match.

Check the value of the OVO managed node OvCoreID stored in the OVO database using the command:

```
opcnnode -list_id node_list=<OVO managed node>
```

It must match the `<AGENT-COREID>`.

You can change the OVO managed node OvCoreID from the OVO management server using the command:

```
opcnnode -chg_id node_name=<OVO managed node> id=<AGENT-COREID>
```
You can change the OvCoreID on the OVO managed node using the command:

```
ovcoreid -set <NEW-AGENT-COREID>
```

**NOTE**

Changing the OvCoreID of a system is an operation that must be done with great care because it changes the identity of a node. All node-related data, such as messages, are linked by the OvCoreID of a node. Changing the value of the OvCoreID should only be executed by experienced users who know exactly what they want to do and what is being affected by attempting this change, especially on the OVO management server.

5. Check, that all OVO Management Server processes are running using the commands:

```
opcsv -status
```

All registered processes must be in the state `running`.

```
ovc -status
```

All registered core processes must be in state `running`.

6. Make sure that the operator is responsible for the:

- OVO managed node and its node group
- Message group

Reload the Message Browser.

7. Check for pending certificate requests.

On the Certificate Authority server enter the command:

```
opccsa -list_pending -l
```

Check if the OVO managed node is listed by nodename, IP address or OvCoreID and whether all parameters are consistent.

Manually grant pending certificate requests with the command:

```
opccsa -grant <NODE> | <COREID>
```

If the parameter are not consistent, change the values on the OVO management server and OVO managed node, as required.
On the OVO managed node, stop and restart all processes with the commands:

`ovc -kill`

Verify, that all processes are stopped with the command:

`ps <OPT> | grep /opt/OV`

`ovc -start`

NOTE

To manually trigger a Certificate Request, first check that there is no certificate already installed with the command:

`ovcert -status`

If no certificate is installed, enter the command:

`ovcert -certreq`

If a certificate is already installed, the following error message is displayed:

ERROR: (sec.cm.client-125) There is already a valid certificate for this node installed.

8. If there are no OVO managed node messages in the Message Browser on OVO managed node, execute the following checks:

- Check if all processes are running:
  
  `ovc -status`

  All registered processes must be running and no process should run twice.

- Check if the expected policies are deployed:
  
  `ovpolicy -list`

- Check the MANAGER, MANAGER_ID, and CERTIFICATE_SERVER settings:
  
  `ovconfget sec.cm.client CERTIFICATE_SERVER`

  This must match the Certificate Authority server.

  `ovconfget sec.core.auth MANAGER`

  This must match the OVO management server.
ovconfget sec.core.auth MANAGER_ID
This must match the OvCoreID of the OVO management server.

ovconfget eaagt OPC_PRIMARY_MGR
This setting is optional, but when set, it must match the OVO management server.

---

**NOTE**

If the OVO management server is not the primary manager, additional checks have to be performed.

---

The OVO management server must appear with consistent values in the file:

```
<DATADIR>/datafiles/policies/mgrconf/<ID>_data
```

- Check the settings of message suppression.
- Check the settings of message buffering.
- Check if the message buffer file is growing:
  ```
  ls -l <DATADIR>/tmp/OpC/msgagtdf
  ```
  or on OVO management server:
  ```
  opcragt -status <nodename>
  ```
- Send a message to be forwarded to the server:
  ```
  opcmsg a=appl o=object msg_t=<my_text>
  ```
- Check if messages appear in the message manager queue file:
  ```
  strings /var/opt/OV/share/tmp/OpC/mgmt_sv/msgmgrq | grep <my_text>
  ```

9. If **DEPLOYMENT**, **ACTIONS** or **HBP** to an OVO managed node fails, on the OVO managed node, check the status of the agent with the command:

```
opcragt -status
```

If this reports no problems, the problem is not HTTPS communication dependent.
Problems during Certificate Deployment

During certificate deployment, the situation may arise that there are two pending certificate requests for the same node in the Certificate Server Adapter's list of pending certificate requests.

For example, this can occur if the certificate request is triggered from the node. This certificate request is not granted and remains pending in the Certificate Server Adapter's internal list. If you now de-install the agent software and re-install it, another certificate request is triggered. The new request also contains a new OvCoreID, because re-installing the node generates a new OvCoreID. This certificate also remains in the list of pending certificate requests.

The listing of the pending certificate requests also contain a time stamp of when the certificate request was received by the OVO management server. It is clear which certificate request is newer and valid. Grant the newest one and remove any older requests.

Alternatively, there are two further ways of removing unwanted certificate requests:

- Log in as an OVO administrator and remove all certificate requests for a “problematic” node and then issue a new certificate request with the command:

  ovcert -certreq

  This results in a single certificate request for the node which can then be mapped and granted in the usual way. See Chapter 5, “Working with HTTPS Nodes,” on page 99.

- If as administrator, you cannot execute the ovcert -certreq command on the node and so cannot issue a new certificate request, then retrieve the valid OvCoreID from the node by executing the command:

  `<OvInstallDir>/bin/bbcutil -ping <nodename>`

  List all certificate requests and grant the certificate request that contains valid OvCoreID and remove any others.
Invalid OvCoreIds on OVO Management Servers

After reinstalling an OVO management server, a new OvCoreId is assigned. However, it may happen that the OvCoreId that was used to sign existing policies is invalid. The new OVO management server OvCoreId is used to verify the signature while loading policies on a managed node. Since the policies are signed by the old OVO management server OvCoreId, you see error message from interceptors which informs that verification failed.

NOTE
Policies that are locally deployed to the OVO management server node may also contain the wrong signature.

For example, opcmsgi and opcmona fail to read their policies, but opcle is successful.

An error message of the following form may be displayed:
40-1867 Cannot validate policy signature.

To correct this type of situation, execute the following commands.

On the OVO management server system:

1. Delete the contents of the directory:
   ```
   /etc/opt/OV/share/conf/OpC/mgmt_sv/templates/utf8/ux_compress
   ```

2. Remove all existing policies:
   ```
   ovpolicy -remove -all
   ```
   or manually delete all files from the directory:
   ```
   /var/opt/OV/datafiles/policies
   ```

3. Stop all OpenView processes:
   ```
   ovc -kill
   ```

4. Check the MANAGER_ID values with the following commands. An example of the expected output is also shown:
   ```
   ovcoreid
   edb78a08-1431-74ff-17c1-f4aef838aa2b
   opcnode -list_id node_list="<mgmt_srv_nodename>"
   ```
Troubleshooting HTTPS-based Communication

Troubleshooting

List of IDs for node(s):
Name = <nodename> ID = edb78a08-1431-74f4ae838aa2b

```
$ ovcert -list
+------------------------------------------------------+
  | Keystore Content                                    |
+------------------------------------------------------+
  | Certificates:                                        |
  |   edb78a08-1431-74ff-17c1-f4ae838aa2b (*)            |
+------------------------------------------------------+
  | Trusted Certificates:                               |
  |   CA_edb78a08-1431-74ff-17c1-f4ae838aa2b            |
+------------------------------------------------------+
```

These should all return the same value for the MANAGER_ID.

If the values are not the same, take the value of MANAGER_ID returned by the ovcert -list command and update the incorrect value as follows:

- If the value of MANAGER_ID returned by the ovcoreid command does not match the value returned by the ovcert -list command, reset the OvCoreId of the OVO management server with the command:
  ```
  ovconfchg -ns sec.core.auth -set MANAGER_ID <mgmt_srv_coreid>
  ```

- If the value of MANAGER_ID returned by the ovcoreid command does not match the value returned by the opcnode -list_* command, reset the OvCoreId of the OVO management server with the command:
  ```
  opcnode -chg_id node_name="<mgmt_srv_nodename>"
  ```

5. ovc -start

On the managed node (when the managed node is not the OVO management server system):

1. Remove all existing policies:

  ```
  ovpolicy -remove -all
  ```

  or manually delete all files from the directory:

  ```
  /var/opt/OV/datafiles/policies
  ```
2. Stop all OpenView processes:
   
   `ovc -kill`

3. List all existing certificates:
   
   `ovcert -list`

   +------------------------------------------------------+
   | Keystore Content                                       |
   +------------------------------------------------------+
   | Certificates:                                          |
   | adb66a06-1666-23aa-18b1-e4dcf454bb3a (*)                |
   +------------------------------------------------------+
   | Trusted Certificates:                                 |
   | CA_edb78a08-1431-74ff-17c1-f4ae838aa2b                 |
   +------------------------------------------------------+

4. Remove all certificates listed in the previous step:
   
   `ovcert -remove <cert_id>`

5. Set the value for `MANAGER_ID` on this managed node to the OvCoreId of the OVO management server. This value was returned by the `ovcoreid` command on the OVO management server above:
   
   `ovconfchg -ns sec.core.auth -set MANAGER_ID \<mgmt_srv_OvCoreId>`

6. Start the OpenView processes:
   
   `ovc -start`

   A new certificate request should be made automatically as the existing certificates have been removed. Check this from the OVO management server GUI.

On the OVO management server system:

1. With the OVO management server GUI, check for and, if necessary, grant the pending certificate request for the managed node in question.

   You should see a message of the form:
   
   `Certificate was successfully installed on the node.`

2. To be able to re-deploy the `nodeinfo` policy, from the OVO Administrators GUI, temporarily modify node settings for the OVO management server and the OVO managed nodes in question.
You may change it back later.

    opcragt -distrib -templates -force <mgmt_Server hostname>
opcragt -distrib -templates -force <node hostname>

3. If required, return the node settings to their original values and trigger template distribution on the OVO management server and affected managed nodes.

Certificate Backup and Recovery in OVO

It is extremely important to be aware of the impacts of losing a private key or when keys and certificate errors arise. The normal configuration upload and download does not include certificate and key data.

There is a utility on the OVO management server to backup and recover certificates plus the associated private keys and core IDs:

    /opt/OV/bin/OpC/opcsvcertbackup/

This utility has the following options:

- **--remove**
  - Removes all certificates from an OVO management server, including:
    - Certificate Authority root certificate and its private key.
    - Server certificate and its private key.
    - Node certificate on the OVO management server.
  - However, a backup is also created automatically before the removal takes place.

- **--backup**
  - A tar archive is created at the following default address:
    /tmp/opcsvcertbackup.<date_time>.tar
  - The <date_time> format is YYMMDD_hhmmss.
  - The default storage location can be changed by using the --file option.
  - The information recorded includes:
    - Certificate Authority root certificate, private key and ID
    - OVO management server certificate with key and core ID
— Node certificate with key and core ID

You must secure the data by using the `–pass` option with a password.

The tar archive contains a text file named:

`opcsvcertbackup.<date_time>.txt`

This information can be useful for archiving and includes OvCoreIds of the backed up certificates, hostname, and time stamp of the backup. This information is not used during a restore.

- **–restore**

  A tar archive as created using the `–backup` option can be restored using this command.

  The filename must be provided with the `–file` option. The password used at backup time must be entered with the `–pass` option.

  The restore cannot work, if any of the certificates or private keys for the Certificate Authority, OVO management server, or node already exists on the OVO management server system but are not the same as the corresponding values stored in the backup archive.

  To avoid this, enforce the restore by using the `–force` option.

  `opcsvcertbackup` also returns with an error when the OvCoreIds of the certificates to be restored do not fit with those stored in the OVO database. When the `–force` option is used, the OvCoreIds are replaced and confirmation is displayed.

**When to Backup Certificates**

The following are the times when a backup using `opcsvcertbackup` is recommended:

- **Initial OVO Installation**

  After a successful OVO management server installation, it is highly recommended to make a backup of the certificate data with the command:

  `opcsvcertbackup –backup`

  The resulting tar archive should be stored in a secure place.
Troubleshooting HTTPS-based Communication

Troubleshooting

- **OVO Management Server Re-installation on Alternative System**
  
  Perform a standard OVO management server installation on the alternative system. Install the backup from the original OVO management server installation onto the newly installed system with the command:

  `opcsvcertbackup -restore -file <filename> -pass <password> -force`

---

**NOTE**

The `-force` option must be used because the server installation has automatically created a Certificate Authority, OVO management server, and node certificates. These certificates are unsuitable because the managed nodes are configured to use the existing ones from the first installation.

---

- **Recovery**
  
  If something is deleted accidentally, use the command:

  `opcsvcertbackup -restore -file <filename> -pass <password>`

  Carefully check any error output.

- **Recovery from Configuration Errors**
  
  If a normal recovery without force option is not successful, check the error messages from the `opcsvcertbackup` call. If this does not help, clean the certificate information stuff with the command:

  `opcsvcertbackup -remove`

  or directly overwrite the existing certificate configuration with the command:

  `opcsvcertbackup -restore -file <filename> -pass <password> -force`

- **Configuring a Certificate Trust for MoM Environments**
  
  After creating a certificate trust it is recommended that you make a new backup. This ensures that the additional root certificate(s) can be restored in case a recovery is needed.
• **Configuring a Shared Certificate Authority**

  When configuring a shared Certificate Authority, the following command can be useful for removing the unwanted certificates from a second OVO management server installation.

  `opcsvcertbackup --remove`

  For further details “Environments Hosting Several Certificate Servers” on page 55.
B Configuring HTTPS-based Communication
Communication Configuration Parameters

HP OpenView applications may be customized for an installation using configuration parameters. The communication broker configuration parameters are contained in the bbc.ini file located at the following address:

<OVDataDir>/conf/confpar/bbc.ini

The parameters used for communication are described in “HTTPS Communication Configuration File” on page 218.

The Communication Broker uses the namespace bbc.cb. An additional namespace, bbc.cb.ports, has been defined to specify the Communication Broker port number for all nodes. This enables different Communication Brokers to have different port numbers. This configuration takes precedence over the SERVER_PORT parameter defined in the namespace bbc.cb.

NOTE

A namespace is a unique URL (Uniform Resource Locator).

For example:

www.anyco.com or abc.xyz

Namespaces provide a simple method for qualifying element and attribute names used in Extensible Markup Language documents by associating them with namespaces identified by URL references.

The name/value pairs in the bbc.cb.ports namespace define the port numbers for the Communication Brokers within the network. The syntax of the name/value pairs is:

NAME=<host>:<port> or NAME=<domain>:<port>

Multiple host/port or domain/port combinations may be defined per line. Each is separated by a comma or semicolon.

A domain takes the form *.domainname. All entries for this domain will use the specified port. More specific entries take precedence. The name of the name/value pair is ignored, although the names must be unique within this namespace. The following are entry examples:
• HP=jago.sales.hp.com:1383, *.sales.hp.com:1384; *.hp.com:1385
• SUN= *.sun.com:1500

In this example the Communication Broker running on the host jago.sales.hp.com will have the port number 1383. All other hosts within the domain sales.hp.com use the port number 1384. All other hosts within the domain hp.com use the port number 1385. Hosts in the domain sun.com use the port number 1500. All other hosts use the default port number 383.
HTTPS Communication Configuration File

bbc.ini(4)

NAME

bbc.ini – Configuration file for HTTPS communication.

DESCRIPTION

bbc.ini is the configuration file of an OVO managed node using HTTPS communication and is located at:

/<OVDataDir>/conf/confpar

It consists of sections headed by namespaces which contain the settings for each namespace. The bbc.ini file contains the namespaces listed below. Possible and default settings are described for each namespace.

bbc.cb

The Communication-Broker Namespace. You can use the following parameters:

string CHROOT_PATH = <path>

On UNIX systems only, the chroot path is used by the ovbbcccb process. If this parameter is set, the ovbbcccb process uses this path as the effective root thus restricting access to a limited part of the file system. Default is <OvDataDir>. This parameter is ignored on MS Windows and Sun Solaris 7 systems. See the chroot man page for details on chroot.

bool SSL_REQUIRED = false

If this parameter is set to true, the communication broker requires SSL authentication for all administration connections to the communication broker. If this parameter is set to false, non-SSL connections are allowed to the communication broker.

bool LOCALCONTROL_ONLY = false
If this parameter is set to true, the communication broker only allows local connections to execute administrative commands such as start and stop.

```python
bool LOG_SERVER_ACCESS = false
```

If this parameter is set to true, every access to the server is logged providing information about the sender's IP address, requested HTTP address, requested HTTP method, and response status.

```python
int SERVER_PORT = 383
```

By default this port is set to 383. This is the port used by the communication broker to listen for requests. If a port is set in the namespace [bbc.cb.ports], it takes precedence over this parameter.

```python
string SERVER_BIND_ADDR = <address>
```

Bind address for the server port. Default is INADDR_ANY.

**bbc.cb.ports**

The Communication-Broker-Port Namespace. This parameter defines the list of ports for all Communications Brokers in the network that may be contacted by applications on this host. The default port number for all communication brokers is 383. You can use the following parameters:

```python
string PORTS
```

This configuration parameter must be the same on all nodes. To change the port number of a communication broker on a particular host, the hostname must be added to this parameter, e.g. `name.hp.com:8000`. You can use an asterisk “*” as a wild card to denote an entire network, e.g., `*.hp.com:8001`. Note too, that either a comma “,” or a semi-colon “;” should be used to separate entries in a list of hostnames, for example;

```
```

In these examples, all hostnames ending in “hp.com” will configure their BBC Communication Broker to use port 8001 except host “name” which will use port 8000. All other hosts use the default port 383.
You can also use IP addresses and the asterisk wildcard (*) to specify hosts. For example:

```
15.0.0.1:8002, 15.*.*.*:8003
```

### bbc.http

The HTTP Namespace for node-specific configuration. For application-specific settings, see the section bbc.http.ext.*. Note that application-specific settings in bbc.http.ext.* override node-specific settings in bbc.http. You can use the following parameters:

#### int SERVER_PORT = 0

By default this port is set to 0. If set to 0, the operating system assigns the first available port number. This is the port used by the application `<appName>` to listen for requests. Note that it only really makes sense to explicitly set this parameter in the `bbc.http.ext.<appName>` namespace, as the parameter is application specific with any other value than the default value.

#### string SERVER_BIND_ADDR = <address>

Bind address for the server port. Default is `localhost`.

#### string CLIENT_PORT = 0

Bind port for client requests. This may also be a range of ports, for example 10000-10020. This is the bind port on the originating side of a request. Default is port 0. The operating system will assign the first available port.

Note that MS Windows systems do not immediately release ports for reuse. Therefore on MS Windows systems, this parameter should be a large range.

#### string CLIENT_BIND_ADDR = <address>

Bind address for the client port. Default is `INADDR_ANY`.

#### bool LOG_SERVER_ACCESS = false

If this parameter is set to `true`, every access to the server is logged providing information about the sender’s IP address, requested HTTP address, requested HTTP method, and response status.
string PROXY

Defines which proxy and port to use for a specified hostname.

Format:
proxy:port +{a}-(b);proxy2:port2+{a}-(b); ...;

a: list of hostnames separated by a comma or a semicolon, for which this proxy shall be used.
b: list of hostnames separated by a comma or a semicolon, for which the proxy shall not be used.

The first matching proxy is chosen.

It is also possible to use IP addresses instead of hostnames so 15.*.*.* or 15::*::*::*::*::* would be valid as well, but the correct number of dots or colons MUST be specified. IP version 6 support is not currently available but will be available in the future.

bbc.fx

BBC File-Transfer Namespace for node-specific configuration. For application-specific settings, see the section bbc.fx.ext.*. Note that application-specific settings in bbc.fx.ext.* override node-specific settings in bbc.fx. You can use the following parameters:

int FX_MAX_RETRIES = 3

Maximum number of retries to be attempted for the successful transfer of the object.

string FX_BASE_DIRECTORY = <directory path>

Base directory for which files may be uploaded or downloaded. Default directory is <OvDataDir>.

string FX_TEMP_DIRECTORY = <directory path>

Temporary directory where uploaded files are placed while upload is in progress. At completion of upload, the file will be moved to <directory path>. Default directory is <OvDataDir>/tmp/bbc/fx.

string FX_UPLOAD_DIRECTORY = <directory path>
Target directory for uploaded files. By default this is the base directory. The upload target directory may be overridden with this configuration parameter. Default directory is `FX_BASE_DIRECTORY`.

**bbc.snf**

BBC Store-and-Forward Namespace for node-specific configuration. For application-specific settings, see the section `bbc.snf.ext.*`. Note that application-specific settings in `bbc.snf.ext.*` override node-specific settings in `bbc.snf`. You can use the following parameters:

- **string BUFFER_PATH = <path>**
  Specifies the SNF path where the buffered requests are stored. Default is:
  
  `<OVDataDir>/datafiles/bbc/snf/<appName>`

- **int MAX_FILE_BUFFER_SIZE = 0**
  Specifies the maximum amount of disk space that the buffer is allowed to consume on the hard disk.
  
  `0 = No limit`

**bbc.http.ext.***

HTTP External-Communication Namespaces:


See the section `bbc.http` for a list of the parameters you can use in the `bbc.http.ext.*` namespace.

**bbc.fx.ext.***


File Transfer External Namespaces:

`bbc.fx.ext.<compID>.<appName>` and `bbc.fx.ext.<appName>`.
See the section `bbc.fx` for a list of the parameters you can use in the `bbc.fx.ext.*` namespace.

**bbc.snf.ext.**


Store and Forward External Namespace:
`bbc.snf.ext.<compID>.<appName>` and `bbc.snf.ext.<appName>`.

See the section `bbc.snf` for a list of the parameters you can use in the `bbc.snf.ext.*` namespace.

**AUTHOR**

`bbc.ini` was developed by Hewlett-Packard Company.

**EXAMPLES**

PROXY=web-proxy:8088-(*.hp.com)+(*.a.hp.com;*)

The proxy `web-proxy` is used with port 8088 for every server (*) except hosts that match `*.hp.com`, for example, `www.hp.com`. If the hostname matches `*.a.hp.com`, for example, `merlin.a.hp.com` the proxy server will be used.

**SEE ALSO**

`ovbbcpcb (1)`
C HTTPS Communication Architecture
Communication (Broker) Architecture

The Communication Broker acts as a proxy on the local node and provides a central point of entry to the node for all applications on that node. Applications that want to receive data register an address with the Communication Broker. The registration defines the port number, protocol, bind address, and base path the application wants to receive data on. Other applications, local or remote, either query the Communication Broker for the location of the application or use the Communication Broker as a proxy to forward the request to registered applications. The Communication Broker loads configuration data from the standard OpenView Configuration File.

The Communication Broker has the following characteristics:

- The Communication Broker provides a single port solution for the node. Requests for all registered servers on this node can be directed through the Communication Broker. The Communication Broker transparently forwards the request to the registered server in the same way as an HTTP proxy forwards an HTTP request. The default port for the Communication Broker is 383 but can be changed.

- For higher security on UNIX systems, chroot can be used at start up of the Communication Broker. chroot restricts the part of the file system visible to the Communication Broker process by making the specified path act as the root directory, thus reducing exposure to hackers.

- The Communication Broker can be run as non-root on UNIX systems if its port number is greater than 1024.

- The Communication Broker can be configured to run as root-only on UNIX systems to open its port and then switch to a non-root user for all other operations.

- The Communication Broker can be:
  - Started as a daemon on UNIX systems.
  - Installed as a Windows NT Service on Windows systems.

- Control commands for the Communication Broker can be restricted to the local node only.
The Communication Broker applies SSL encryption of data transmission over the network.

The Communication Broker applies SSL authentication through guaranteed identity of senders and receivers.

Figure C-1  Communication Broker Architecture

A Communication Broker configures a minimum of one port for accepting incoming data to a node. The port is associated with an OpenView ID (OVCoreID) to identify the node. The Communication Broker can be configured to open multiple ports for high availability nodes. Each port can have a different identity associated with it. If SSL is enabled, the port is configured with X509 certificates. These certificates allow connecting applications to verify the identity of both message senders and receivers.

All applications on the current node that register with the Communication Broker are automatically registered for all active incoming ports opened by the Communication Broker. The port
associated with the default namespace, bbc.cb, is automatically activated on startup of the Communication Broker. Other ports can be activated or deactivated dynamically after startup. See the command line interface parameters for the Communication Broker for details.
D  Firewalls and HTTPS Communication
Firewall Scenarios

Firewalls are used to protect a company’s networked systems from external attack. They usually separate the Internet from a company’s private intranet. It is also quite common to implement multiple levels of firewalls to restrict access to the more trusted environments from those of lower sensitivity. For example, the research and finance departments may be contained in the environment of highest security, while direct sales may need to be easily accessible from the outside. Systems on the intranet are allowed, under certain conditions, to cross the firewall to access systems on the internet, for example located in the DMZ. The firewall can also allow systems on the Internet to cross the firewall and access systems on the private intranet. For either of these situations, the firewall must be configured to allow that operation.

HP OpenView’s HTTPS communication provides features that allow firewall administrators to configure HP OpenView applications to communicate through firewalls.

Contacting an Application on the Internet from an Intranet using an HTTP Proxy

An HP OpenView HTTPS-based application on a private intranet wants to contact an application outside of the firewall on the public Internet or Demilitarized Zone (DMZ). The OpenView application initiates the transaction and acts as a client contacting a server application on the Internet. The server application could be another HP OpenView application acting as an HTTP server or any other HTTP server application. A common example of a client is a web browser on the private intranet wanting to contact a web server on the Internet. An HTTP proxy must be configured in the browser which forwards the request across the firewall and contacts the web server in the Internet. The firewall is configured to allow the HTTP proxy to cross the firewall. The firewall does not allow the web browser to directly cross the firewall. In the same way, HP OpenView’s HTTPS communication applications can also be configured to use HTTP proxies to cross firewalls.
Contacting an Application on the Internet from an Intranet without an HTTP Proxy

An HP OpenView HTTPS-based application on a private intranet wants to contact an application outside of the firewall on the Internet without using an HTTP proxy. The firewall must be configured to allow the HP OpenView application on the private intranet to cross the firewall. This is very similar to configuring a firewall to allow an HTTP proxy to cross the firewall. The firewall administrator may want to set source and target ports for the transaction to restrict communication across the firewall. The \texttt{CLIENT\_PORT} configuration parameter specifying the source ports can be set from the HP OpenView application when initiating the transaction. The target or destination port is defined in the URL (Uniform Resource Locator or Identifiers) address used to contact the HTTP server on the Intranet. This is the communication broker port on the target node.

Contacting an Application within a Private Intranet from an OpenView Application on the Internet

An HP OpenView HTTPS-based application on the Internet wants to contact an application on a private intranet. This means that a firewall must be crossed from the outside and is usually only allowed by organizations under very restricted conditions set by the firewall administrator. The initiating or client application may do this using an HTTP proxy or go directly through the firewall. The HTTP proxy is outside the firewall and the firewall must be configured to allow the HTTP proxy to cross it. The HTTP proxy could either directly contact the server on the private intranet or go through another proxy, in a cascading proxies arrangement. In either case, the HP OpenView’s HTTPS communication client application is configured in the same way. However, the HTTP proxies must be configured differently.

Contacting an Application within a Private Intranet from an OpenView Application on the Internet without using HTTP Proxies

An HP OpenView HTTPS-based application on the Internet wants to contact an application on the private intranet, but there is no HTTP proxy. The firewall must be configured to allow the HP OpenView client application to cross the firewall. The firewall administrators may want to
set the source and target ports for the transaction to restrict communication across the firewall. The `CLIENT_PORT` configuration parameter specifying the source port can be set from the HP OpenView application when initiating the transaction. The target or destination port used to contact the HTTP server on the Intranet is defined in the URL address and is the Communication Broker port on the target node.

If the target server is registered with the Communication Broker, the target port will always have the port number of the Communication Broker. This makes it easier when configuring firewalls. It can greatly reduce the number of target ports an administrator must configure at the firewall.
Appendix E

E OVO 8.1 Quick Start Guide
OVO 8.1 Quick Start for OVO 7.x Users

Table E-1 gives you a concise overview of the new commands in OVO 8.1 and their counterparts in OVO 7.1. For full details about any command, refer to the command man page.

The most important new OVO 8.1 commands are introduced in “HTTPS Communication Administration Commands in OVO” on page 37.

NOTE

The wrapper utilities such as opcagt, and opctemplate, do NOT provide the same output format as the DCE-based opcxxx commands.

Table E-1

<table>
<thead>
<tr>
<th>OVO 7.x Command</th>
<th>OVO 8.1 Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>opcagt</td>
<td>ovc</td>
</tr>
<tr>
<td>-help</td>
<td>ovc -help</td>
</tr>
<tr>
<td>-start</td>
<td>ovc -start AGENT</td>
</tr>
<tr>
<td></td>
<td>ovc -restart AGENT</td>
</tr>
<tr>
<td>-stop</td>
<td>ovc -stop</td>
</tr>
<tr>
<td>-status</td>
<td>ovc -status</td>
</tr>
<tr>
<td>-kill</td>
<td>ovc -kill</td>
</tr>
<tr>
<td>-trace</td>
<td>ovc -trace</td>
</tr>
<tr>
<td>-version</td>
<td>ovc -version</td>
</tr>
<tr>
<td>opcagt</td>
<td>ovdeploy, ovconfpar</td>
</tr>
<tr>
<td>-agent_version</td>
<td>ovdeploy -inv -host &lt;node&gt;</td>
</tr>
<tr>
<td>-get_config_var</td>
<td>ovconfpar -get</td>
</tr>
<tr>
<td>-set_config_var</td>
<td>ovconfpar -set</td>
</tr>
</tbody>
</table>
Table E-1  Command Mapping Table between OVO 7.x and OVO 8.1

<table>
<thead>
<tr>
<th>OVO 7.x Command</th>
<th>OVO 8.1 Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>opctemplate</td>
<td>ovpolicy</td>
</tr>
<tr>
<td>-help</td>
<td>ovpolicy -help</td>
</tr>
<tr>
<td>-l</td>
<td>ovpolicy -list</td>
</tr>
<tr>
<td>-e</td>
<td>ovpolicy -enable</td>
</tr>
<tr>
<td>-d</td>
<td>ovpolicy -disable</td>
</tr>
<tr>
<td>opcsv</td>
<td>ovc</td>
</tr>
<tr>
<td>-help</td>
<td>ovc -help</td>
</tr>
<tr>
<td>-start</td>
<td>ovc -start SERVER</td>
</tr>
<tr>
<td></td>
<td>ovc -restart SERVER</td>
</tr>
<tr>
<td>-stop</td>
<td>ovc -stop</td>
</tr>
<tr>
<td>-status</td>
<td>ovc -status</td>
</tr>
<tr>
<td>-trace</td>
<td>ovc -trace</td>
</tr>
<tr>
<td>opctransm</td>
<td>ovdeploy (HTTPS Agents)</td>
</tr>
<tr>
<td></td>
<td>opctransm (DCE Agents)</td>
</tr>
</tbody>
</table>
Master Index

Symbols
<$#> variable, AR:165
<$*> variable, AR:165
<$\>+1> variable, AR:165
<$\>+2> variable, AR:166
<$\>-1> variable, AR:165
<$\>-2> variable, AR:166
<$\>-n> variable, AR:166
<$@> variable, AR:165

Numerics
<$1> variable
  logfiles, AR:162
  SNMP traps, AR:165

A
A message attribute, AR:76
<$A> variable, AR:166
aa* temporary file, AR:359
About Virtual Terminal, DCE:174
access
  See also accessing
  file permissions, AR:463
  remote, AR:467
  restrictions, CG:56
  terminal, CG:226
accessing
  See also access
  files, CG:226
GUI
  administrator, AR:464
  Java, AR:465
  Motif, AR:464
Jovw, AR:336–AR:338
man pages
  command line, AR:557
  HTML format, AR:557
managed node MIB, AR:433–AR:434
NNM, AR:328–AR:330
OpenView applications, CG:156
OVO, AR:462
programs
  HP-UX, AR:465
  MPE/iX, AR:465
quick filters, CG:214
terminal, CG:177
account, primary, AR:468

acknowledgements
  See also acknowledging messages;
  messages
  annotating, CG:366
  automatic, CG:166
  description, CG:183
  reviewing, CG:184
acknowledging messages
  See also acknowledgements; messages
  escalated messages, CG:453
  message keys, CG:365
  notification messages, CG:475
ACL Info application, DCE:498
actagtq pipe file, AR:358
actagtq queue file, AR:358
action
  See also actions
  agents, AR:255
  variables, AR:160–AR:161
Action Report, AR:110
action-allowed managers
  configuring, CG:459
  specifying, CG:469
ACTIONALLOWMANAGERS keyword, AR:120
actions
  See also action
  applying to all nodes in hierarchy,
    CG:233–CG:234
  automatic, CG:51–CG:52
  centralizing, CG:305
  control-switched messages, CG:474
  enabling on secondaring manager, CG:468
  evaluating results, CG:164
  integrating applications as, AR:255–AR:256
  operator-initiated, CG:53–CG:54
  overview, CG:51–CG:54
  protecting, AR:471–AR:474
  responding to messages, CG:393
  scheduled, AR:169
  stderr, CG:164
  stdout, CG:164
  verifying
    automatic, CG:165–CG:166
    operator-initiated, CG:167
Actions policy, CG:134
activating
managed nodes
  AIX, DCE:43–DCE:45
  HP-UX, DCE:93–DCE:96, DCE:331–DCE:333
active message browser
  See also filtered message browser; history message browser; message browser; pending messages browser
figure, CG:92
  overview, CG:96–CG:97
actreqp pipe file, AR:353
actreqq queue file, AR:353
actrespp pipe file, AR:353
actrespq queue file, AR:353
Adapters application, DCE:209
Add Configuration window, CG:314
Add MPE/iX Console Messages window, CG:423
Add Node for External Events window, CG:236
Add SNMP Trap window, CG:418
adding
  annotations, CG:179
  message groups, CG:252, AR:73
  nodes to OVO, CG:236–CG:248
  external nodes, CG:238
  from IP submaps, CG:241
  from OVO Add Node window, CG:242–CG:245
  internal nodes, CG:236
  methods, CG:229
  node groups, AR:71
  with templates, CG:314
OVO variables, CG:174
SNMP trap templates, CG:418
tabs to browser pane, CG:214
administrative rights
  See also OVO administrator
administrator. See template administrators; OVO administrator
administrator-defined defaults, CG:191
advanced options
  message conditions, CG:408
  MPE/iX console messages, CG:424
advantages
  backups
    automatic, AR:490
    offline, AR:489
  flexible management, CG:447
  operator message browser, CG:223
  OVKey licenses, AR:510
template groups, CG:310
agdobserver monitor template, AR:221
agent accounts
agent filesets in OVOPC-CLT
  English-only, DCE:82
generic, DCE:82
agent profile
  alternative users, HTTPS:77
  patching, HTTPS:80
  sudo, HTTPS:81
upgrading, HTTPS:80
agents. See action agents; OVO agents
AIX managed nodes
  DCE
    configuring, DCE:40–DCE:41
    requirements, DCE:36
HACMP
  installing agents, DCE:52–DCE:53
  resetting IP, DCE:50
NCS requirements, DCE:36
OVO
  activating, DCE:43–DCE:45
  default operator, DCE:61
de-installing agents, DCE:54
directory structure, DCE:60
file locations, DCE:60
hardware requirements, DCE:33
include file, DCE:63
installation requirements,
  DCE:33–DCE:36
installation tips, DCE:37–DCE:39
installing agents, DCE:42–DCE:45
libraries, DCE:62–DCE:64
logfile locations, AR:508
makefile, DCE:64
organization, DCE:60–DCE:61
overview, DCE:31–DCE:65
preconfigured elements, DCE:55–DCE:57
removing agents, DCE:54
scripts and programs, DCE:58–DCE:59
Smit User Interface, DCE:57
SNMP event interceptor, DCE:56
software requirements, DCE:33–DCE:36
system resource files, DCE:61
troubleshooting IP aliases,
  DCE:49–DCE:50
OVPA, AR:207
alarmgen monitor template, AR:221
All Active Details Report, AR:114
All Active Messages Report, AR:110, AR:114
All History Details Report, AR:114
All History Messages Report, AR:114
All Pending Details Report, AR:114
All Pending Messages Report, AR:114
alternative accounts
alternative users, HTTPS:70
agent profile, HTTPS:77
changing default port, HTTPS:76
comparison with DCE agents, HTTPS:84
configuring the management server,
  HTTPS:75
installation, HTTPS:73
limitations, HTTPS:71
patching, HTTPS:80
preparation, HTTPS:72
sudo, HTTPS:81
upgrading, HTTPS:80
analyzing
data with OVPA, AR:208
symptoms in OVO, AR:379
annotating
acknowledgements, CG:366
messages
  escalated, CG:454
  notification, CG:475
notations
  overview, CG:179–CG:181
  reviewing, CG:164–CG:165
APIs
  man pages
    Developer’s Kit, AR:564
    OVO, AR:562
managed nodes, AR:543
message, CG:391–CG:392
MSI, AR:260
Novell NetWare, DCE:220–DCE:221
opcmsg (3), DCE:113
application
  ping, HTTPS:184
registered with communication broker,
  HTTPS:185
status, HTTPS:185
Application Desktop window, CG:60
Application message attribute, AR:77
applications
  accessing OpenView, CG:156
  assigning to operators, AR:245
  Broadcast, CG:164
  Citrix MetaFrame, DCE:438–DCE:441
configuring templates, CG:329
customizing, CG:171
HP-UX
  ASCII SAM, DCE:101
  EMS Resources, DCE:118–DCE:119
  Motif SAM, DCE:101
integrating into OVO
  actions, AR:255–AR:256
  Application Desktop, AR:246–AR:247
  broadcast command, AR:254
  components, AR:245
  Ethernet Traffic HP as an OV application,
    AR:250
  HP applications, AR:245
  monitoring applications, AR:257
  OpenView plug-in, AR:246
  overview, AR:243–AR:262
  OVO applications, AR:246
intercepting messages, AR:259
Java GUI
  comparisons, AR:318
  OpenView, AR:330–AR:332
  monitoring logfiles, AR:258
  Motif GUI, AR:318
  MPE/iX, DCE:172–DCE:174
Novell NetWare
  NetWare Tools, DCE:209–DCE:212
  NMA, DCE:212–DCE:214
  overview, DCE:204–DCE:214
OVO
  description, CG:54
  types, CG:235
  OVPA, AR:218
solving problems, CG:170–CG:171
SSP Tools, DCE:306
starting, CG:170

AR    OVO Administrator’s Reference
DCE    OVO DCE Agent Concepts and Configuration Guide
CG    OVO Concepts Guide
HTTPS    OVO HTTPS Agent Concepts and Configuration Guide
### Master Index

| accounts, AR:466                  | messages, AR:75–AR:77          |
| I/O, AR:467                      | MPE/iX console message templates defaults, CG:424 |
| remotely, AR:467                 | auditing, CG:226               |
| tailored set, CG:207             | levels, AR:475–AR:478          |
| variables, AR:171–AR:186         | modes, AR:475                  |
| Applications folder              | Auditlog application, DCE:438  |
| figure, CG:75                    | configuring DCE nodes to use authenticated RPCs, AR:454 |
| overview, CG:75                  | automatic backups              |
| applying actions to all nodes in hierarchy, CG:233–CG:234 | advantages, AR:490 |
| architecture                     | disadvantages, AR:491          |
| communication broker, HTTPS:228  | excluding files                |
| HTTPS agent, HTTPS:27           | database, AR:491               |
| OVO in a Cluster environment, DCE:451 | temporary, AR:491            |
| archive log mode                 | recovering configuration data, AR:498–AR:500 |
| database                         | automatic de-installation      |
| description, AR:491              | See also de-installing         |
| enabling, AR:492–AR:493          | AIX, DCE:54                    |
| description, AR:488              | HP-UX, DCE:96                  |
| ARPA hostnames, mapping to NS node names, DCE:178–DCE:181 | Linux, DCE:140 |
| ASCII character sets, AR:291     | automatic installation         |
| ASCII SAM, DCE:101               | See also installing            |
| assigning                        | AIX, DCE:42                    |
| applications to operators, AR:245 | automating standard scenarios, CG:364 |
| passwords                        | avoiding duplicate messages, CG:417 |
| managed nodes, AR:468–AR:470     | B                                |
| MPE/iX, AR:469                   | backing up data on management server, AR:488–AR:500 |
| Novell NetWare, AR:470           | backup                          |
| UNIX, AR:469                     |                                  |
| Windows NT, AR:470               |                                  |
| templates                        |                                  |
| distributing, CG:315             |                                  |
| managed nodes, CG:313            |                                  |
| overview, CG:313–CG:315          |                                  |
| attributes                       |                                  |
| custom message                   |                                  |
| overview, CG:147                 |                                  |
| viewing, CG:148                   |                                  |
| message                          |                                  |
| examining, CG:144                 |                                  |
| modifying, CG:145                 |                                  |
| message forwarding, CG:449       |                                  |
| message forwarding templates, AR:138 |                                  |
Master Index

certificate, HTTPS:212
Backup message group, AR:72
backups
  automatic, AR:490–AR:497
  recovering configuration data, AR:498–AR:500
offline, AR:489
server, CG:469
tools, AR:488
backup-server template, AR:117
Bad Logs (10.x/11.x HP-UX) logfile, DCE:98
bbci.ini configuration file, HTTPS:220
bbcutil, HTTPS:37
benefits, OVO, CG:33
binaries
  common, AR:190
  customized, AR:191
  filenames, AR:194
Boot the NetWare Server (NCF) application, DCE:209
Bound Protocols application, DCE:209
Broadcast application, CG:164, DCE:172
broadcast commands
  integrating applications, AR:254
  starting
    on managed nodes, AR:261–AR:262
    remotely, AR:467
broadcasting commands
  overview, CG:175–CG:176
browser pane
  adding tabs, CG:214
  figures
    disabled, CG:203
    main window, CG:89
    message browser, CG:90
    popup menu, CG:115
  hiding, CG:203
  overview, CG:89–CG:91
  popup menus, CG:115
Browser Settings dialog box
  figure, CG:213
browsing messages effectively,
  CG:134–CG:138
buffering messages
  description, CG:37
  parameters, AR:132
  service hours, CG:439
building managed nodes, CG:227
Bull DPX/20, DCE:59

C
<$C> variable, AR:166
C2 security
  techniques, CG:226
Cancel Reboot application, DCE:394
case-sensitivity in pattern-matching, CG:339
catalogue, message, CG:318
  central
    competence centers, CG:450–CG:451
    management server
      action-allowed manager, CG:459
      configuring, CG:462
      description, CG:459
      secondary manager, CG:460
centralizing actions, CG:305
Cert. State Overview, AR:112
certificate
  backup, HTTPS:212
  opcsvcertbackup, HTTPS:212
  restore, HTTPS:212
certificate client, HTTPS:48, HTTPS:53
certificate server, HTTPS:48, HTTPS:52
  merging, HTTPS:56
  multiple, HTTPS:55, HTTPS:59
  sharing, HTTPS:61
certificates, HTTPS:51
  add node to node bank, HTTPS:162
  creating, HTTPS:155
  delete request, HTTPS:161
  deny, HTTPS:161
  deploying automatically, HTTPS:158
  deployment troubleshooting, HTTPS:208
  distributing, HTTPS:155
  generation, HTTPS:164
  grant request, HTTPS:161
  hostname, HTTPS:156
  installation key, HTTPS:169
  IP address, HTTPS:156
  managing, HTTPS:161
  manual deployment, HTTPS:169
  map to selected node, HTTPS:163
  mapped to, HTTPS:156
  OvCoreID, HTTPS:156
  platform, HTTPS:157
  requests window, HTTPS:156
select all mapped requests, HTTPS:162
select all unknown nodes, HTTPS:162
troubleshooting, HTTPS:199
troubleshooting OvCoreIds, HTTPS:209
certification authority, HTTPS:52
cfgchanges file, AR:353
Change Operator Password dialog box figure, CG:186
changing
character set
  logfile encapsulator, AR:291
  managed node, AR:290
communication types, AR:54–AR:56
defaults
  property type of all messages forwarded to OVO, AR:240
  WMI policy name, AR:240
hostnames, AR:514–AR:526
IP addresses, AR:514–AR:526
look and feel of Java GUI, CG:197
operator passwords
  overview, CG:186
OVO administrator responsibility matrix,
  CG:224
passwords, AR:462
refresh interval, CG:193
user names, AR:462
character code conversion, AR:298–AR:304
character sets
  ASCII, AR:291
  changing
    logfile encapsulator, AR:291
    managed nodes, AR:290
  converting, AR:298–AR:304
English language
  configuring, AR:298–AR:301
  supported, AR:289
  types, AR:292–AR:294
Euro symbol, AR:287
external on managed nodes, AR:291–AR:295
ISO 8859-15, AR:287
Japanese language
  configuring, AR:302–AR:304
  supported, AR:290
  types, AR:294
logfile encapsulator, AR:295–AR:297
Spanish language
  supported, AR:289
charts
  current state, CG:152
  history, CG:154
Check alarmdef application, AR:218
Check parm application, AR:218
choosing web browser, CG:204
Citrix MetaFrame
  applications, DCE:438–DCE:441
  integration
    configuring agent, DCE:434
    configuring server, DCE:435
    ICA Browser service, DCE:435
    installing agent, DCE:434
    logfile templates, DCE:437
    monitored objects, DCE:436
    overview, DCE:433–DCE:437
    Program Neighbourhood service, DCE:436
    software requirements, DCE:433
    versions supported, DCE:433
classifying unmatched messages, CG:49
client-server concept, CG:33–CG:35
cold images, HTTPS:128
closing
  EMS GUI, DCE:116
  messages, CG:178
  cluster, HTTPS:146
Cluster administration
  overview, DCE:449–DCE:465
clusters, mixed, AR:194
CMIP events
  forwarding, CG:416–CG:417
  overview, CG:414–CG:421
coda, CG:398
coda process, AR:355
Cold Boot the NetWare Server (NCF)
  application, DCE:209
collecting messages, CG:319–CG:321
colored_message_lines option
  ito_op, AR:321
  itooprc, AR:323
colors
  figures
    message browser, CG:94
    object pane, CG:140
    shortcut bar, CG:140
message browser, CG:215
Message Groups folder, CG:73
| **AR** | OVO Administrator’s Reference |
| **DCE** | OVO DCE Agent Concepts and Configuration Guide |
| **CG** | OVO Concepts Guide |
| **HTTPS** | OVO HTTPS Agent Concepts and Configuration Guide |

**Master Index**

| messages | changing, CG:94 | locations, CG:139–CG:141 |
| nodes folder, CG:71 |
| columns, message browser | customizing, CG:216 |
| hiding, CG:217 | showing, CG:217 |
| command line | accessing man pages, AR:557 |
| activating OVO agents | AIX, DCE:43 |
| Solaris, DCE:282 |
| interface, AR:136 |
| license maintenance tool, AR:512–AR:513 |
| NNM tools, AR:332 |
| command tracing, AR:67 |
| commands | agent, HTTPS:33 |
| bbcutil, HTTPS:37 |
| broadcasting, CG:175–CG:176 |
| HTTPS communication, HTTPS:37 |
| integrating applications as broadcast, AR:254 |
| opc csa, HTTPS:39 |
| opc csa cm, HTTPS:39 |
| opc ctr lvw, AR:332 |
| op clic | parameters, AR:512–AR:513 |
| syntax, AR:512 |
| op cm ap n ode, AR:332 |
| op c wall, AR:493 |
| ov b ackup.ovp, AR:494–AR:495 |
| ovc, HTTPS:37 |
| ov cetr, HTTPS:39 |
| ov c con fget, HTTPS:37 |
| ov core id, HTTPS:37 |
| ov policy, HTTPS:38 |
| ovrc, HTTPS:38 |
| ov r es tor e.ovpl, AR:495–AR:497 |
| std err, CG:164 |
| std out, CG:164 |
| synchronizing with OVO agent character set, AR:286 |
| communication | competence centers, CG:451 |
| configuration file, HTTPS:220 |
| configuration parameters, HTTPS:218 |
| firewall and internet, HTTPS:233 |
| firewall and proxies, HTTPS:232 |
| firewall scenarios, HTTPS:232 |
| in OVO, HTTPS:26 |
| links | central server configuration, CG:462 |
| manufacturing environment, CG:457 |
| OVO, AR:347–AR:348 |
| OVO troubleshooting, HTTPS:204 |
| software types | changing, AR:54–AR:56 |
| description, AR:39–AR:40 |
| troubleshooting, HTTPS:190, HTTPS:192 |
| communication broker | applications registered, HTTPS:185 |
| architecture, HTTPS:228 |
| community name | opcinfo file, AR:433 |
| SNMP daemon configuration file, AR:434 |
| comparing messages with conditions | match conditions, CG:335–CG:337 |
| pre configured templates, CG:37 |
| competence centers | communication flow, CG:451 |
| configuring, CG:451 |
| distributing responsibility, CG:450–CG:451 |
| overview, CG:450–CG:451 |
| component | embedded performance, CG:398 |
| components | HTTPS agent, HTTPS:27 |
| components in subproducts | English, DCE:83 |
| components, integrating into OVO, AR:245 |
| compression setting types, CG:373 |
| concepts | client-server, CG:33–CG:35 |
| message forwarding, CG:472 |
| trouble ticket system, AR:265 |
| user, CG:55–CG:61 |
| Condition No. window, CG:410 |
| conditions | advanced threshold monitoring, CG:409–CG:410 |
| applying to events, CG:335 |
| match, CG:335–CG:337 |
message
  description, CG:334–CG:337
  overview, CG:330–CG:334
  setting up, CG:333–CG:334
  modifying, CG:338
  multiple for threshold monitoring, CG:411–CG:412
  organizing, CG:337–CG:338
  pattern-matching examples, CG:339–CG:340
regroup
  defining, CG:382
  examples, CG:383
  selecting, CG:338
sequence, CG:355
SNMP trap templates
  defining, CG:418–CG:419
  example, CG:420
specifying for message templates, CG:390
status variables, AR:133
suppress
  deploying, CG:356
  description, CG:334–CG:337
threshhold monitor examples, CG:413
types, CG:338
CONDSTATUSVARS keyword, AR:119
Config alarmdef application, AR:218
Config parm application, AR:218
Config perflbd.rc application, AR:218
Config ttd.conf application, AR:218
configuration
  See also configuring
bbc.ini file, HTTPS:220
communication parameters, HTTPS:218
deployment, HTTPS:31, HTTPS:89
distributing OVO agent to managed nodes, AR:189
downloading data, AR:485–AR:487
file
  distributing, CG:470–CG:471
downloading, CG:470
  responsible manager, CG:463–CG:464
uploading, CG:470
installing on managed nodes,
  AR:187–AR:203
loading default, CG:187–CG:193
protecting distribution, AR:470
push, HTTPS:93
server
  multiple parallel, HTTPS:94
  updating on managed nodes,
    AR:187–AR:203
Configure Management Server window, AR:193
configuring
  See also configuration
application-specific templates, CG:329
automatic acknowledgements, CG:166
basic Distributed Event Interception,
  DCE:100
central server, CG:459
Citrix MetaFrame
  agent, DCE:434
  server, DCE:435
competence centers, CG:451
database on multiple disks, AR:502–AR:503
DCE
  AIX, DCE:40–DCE:41
    managed nodes, AR:452
    management server, AR:452
  SINIX RM/Reliant, DCE:260
    Tru64 UNIX, DCE:326–DCE:327
  ECS event interception, DCE:101
  EMS templates, DCE:120
  escalation policies, CG:453
event correlation, CG:430
filenames on MPE/iX managed nodes,
  DCE:171
filtered message browsers, CG:209
flexible management templates,
  AR:117–AR:153
HTTPS nodes, HTTPS:100
managed nodes
description, CG:38
hierarchies, CG:459
regional management servers,
  CG:461–CG:462
management server
central, CG:462
  English language, AR:298–AR:301
regional, CG:461–CG:462
responsible, CG:463–CG:471
NNM access with command-line tools, AR:332
node
authenticated RPCs, AR:454
DCE cell, AR:454
notification service, AR:268
OpenView Operations for Windows
agents for OVO management server,
AR:232–AR:234
servers to forward messages to OVO,
AR:235–AR:240
OVO
agents for OpenView Operations for
Windows management server,
AR:228–AR:231
elements, CG:219–CG:301
messages forwarded from OpenView
Operations for Windows,
AR:237–AR:239
preconfigured elements, AR:69–AR:186
proxies, HTTPS:140
RPC authentication in OV, AR:458
scheduled outages, CG:442
service hours, CG:442
templates
message forwarding, AR:138
message source, CG:308
multiple, CG:326
threshold monitors, CG:408
time-indifferent templates, CG:466
timeouts for report generation, AR:109
trouble ticket system, AR:269
VantagePoint for Windows
agents on OpenView Operations for
Windows management server, AR:239
Configuring DCE, DCE:40
Connections application, DCE:209
console messages, MPE/ix, CG:422–CG:425
console settings
saving, CG:195–CG:197
consolidating messages in browser, CG:306
continuous message generation, CG:405
core
files, AR:502
follow-the-sun, CG:448–CG:450
managed nodes, CG:228
message
sharing, CG:473
switching, CG:473–CG:474
ccontroller tool, AR:333–AR:334
converting
character sets, AR:298–AR:304
managed node files
EUC, AR:303
ROMANS, AR:300
managed nodes to EUC, AR:306
management server to EUC, AR:305
copying and pasting nodes, CG:242
See also dragging and dropping nodes
corrective actions
automatic, CG:393
managed node, CG:37
operator-initiated, CG:393
Corrective Actions workspace
description, CG:84
evaluating action results, CG:164
correlating
events
description, CG:45, CG:427–CG:428
NNM, CG:431
overview, CG:427–CG:434
messages, CG:359
different sources, CG:429
flexible management environments,
CG:434
managed nodes, CG:429, CG:432
management server, CG:429, CG:433
messages and events, CG:357
counter-based suppression, CG:375
CPU Info application, DCE:210
creating
configuration file
responsible managers, CG:463
message
source templates, CG:309
status, CG:319
mirror online redo logs, AR:503
primary account manually, AR:468
SD-UX depot on remote node,
DCE:87–DCE:88
template
group hierarchies, CG:311
groups, CG:311
Critical message severity level, AR:74
Master Index

Cron (10.x/11.x HP-UX) logfile, DCE: 98
Cron (RedHat Linux) template, DCE: 142
Cron (Solaris) template, DCE: 288
ctrlp pipe file, AR: 353
ctrlq queue file, AR: 353
current state chart
figures
   bar chart, CG: 152
   pie chart, CG: 153
overview, CG: 152
custom message attributes
   adding to your message, CG: 348
   overview, CG: 147
   setting defaults, CG: 324
   viewing, CG: 148
customer-specific sub-tree on management server, DCE: 81
Customize Message Browser Columns dialog box
figures
   Custom tab, CG: 138
   General tab, CG: 137
customized job stream facility
preparing OVO, DCE: 163
setting up on MPE/iX managed nodes, DCE: 162
customizing
applications, CG: 171
binaries, AR: 191
Java GUI, CG: 185
message browser columns
   attributes, CG: 136
   layout, CG: 216
message event notification, CG: 208
operator environment, CG: 185
OVPA, AR: 209
popup menus, CG: 206–CG: 207
reports
   administrator, AR: 113
   operator, AR: 115
scripts, AR: 191
shortcut bar, CG: 204

D
daemons
   DCE
      MPE/iX, DCE: 157
   NCS, DCE: 157
   RPC
      MPE/iX, DCE: 157
      troubleshooting, AR: 427
   SNMP, AR: 434
   SSP snmpd, DCE: 307
data, backing up on management server, AR: 488–AR: 500
database
   archive log mode
      description, AR: 488, AR: 491
      enabling, AR: 492–AR: 493
   configuring on multiple disks,
      AR: 502–AR: 503
   excluding files from automatic backups, AR: 491
group, message target rule example, CG: 465
   improving performance, AR: 371
   maintaining, AR: 501
   moving control files to second disk, AR: 502
   recovering, AR: 498–AR: 499
   removing queue files, AR: 500
   reports, AR: 109–AR: 116
   restoring, AR: 498
   restricting access, AR: 116
   security, AR: 465
tables and tablespaces
   non-OVO, AR: 552
   OVO, AR: 547
troubleshooting, AR: 385–AR: 387
Database message group, AR: 72
Date message attribute, AR: 77
DCE
   changing, AR: 54–AR: 56
   configuring
      AIX, DCE: 40–DCE: 41
      managed nodes, AR: 452
      management server, AR: 452
      SINIX RM/Reliant, DCE: 260
      Tru64 UNIX, DCE: 326–DCE: 327
   description, AR: 39
   nodes
      configuring to run in DCE cell, AR: 454
      configuring to use authenticated RPCs,
      AR: 454
      description, AR: 453
      installing, AR: 453
login failure, AR:468
passwords, AR:467–AR:468
removing
  AIX, DCE:41
  SINIX RM/Reliant, DCE:261
  Tru64 UNIX, DCE:327
security, AR:451–AR:456
servers
description, AR:453
installing, AR:452
DCE agent comparison, HTTPS:31
commands, HTTPS:33
configuration deployment, HTTPS:31
distribution managers, HTTPS:32
multiple parallel configuration servers,
  HTTPS:32
performance, HTTPS:33
processes, HTTPS:34
resource requirements, HTTPS:32
troubleshooting, HTTPS:35
DCE agents
alternative user concept, HTTPS:84
migrate from HTTPS, HTTPS:117
migrate to HTTPS, HTTPS:113
debugging software (de-)installation,
  AR:67–AR:68
Description message attribute, AR:77
def_browser option, AR:321
def_help_url option, AR:323
def_look_and_feel option
  ito_op, AR:321
  itooprc, AR:323
default OVO operator
  AIX, DCE:61
  HP-UX, DCE:108
  Linux, DCE:148–DCE:149
  MPE/iX, DCE:177
Novell NetWare, DCE:218
Sequent DYNIX, DCE:235
SGI IRIX, DCE:249
SINIX RM/Reliant, DCE:267
Solaris, DCE:296
Tru64 UNIX, DCE:348
Windows NT/2000, DCE:430
default_browser option, AR:323
defaults
  assigned by
  administrator, CG:191
  OVO, CG:188
IP map, AR:336
loading configuration, CG:187–CG:193
management server setup, CG:446
message
groups, AR:71–AR:73
mapping on MPE/iX, DCE:165
templates on MPE/iX, CG:424
node groups, AR:71
script and program directory, AR:266
threshold monitor, CG:409
trap and event interception, CG:414
WMI policy name, AR:240
working directory, AR:463
Define Configuration window, CG:313
defining
  conditions
    messages, CG:408
    regroup, CG:382
    SNMP trap templates, CG:418–CG:419
    message groups, CG:50
    report printer, AR:109
    scheduled outages, CG:441
    service hours, CG:440
    templates
      logfiles, CG:388
      messages, CG:389, CG:418
      MPE/iX console messages, CG:423
de-installation
agent software, HTTPS:145
  automatic, HTTPS:145
  manual, HTTPS:145
problems, HTTPS:145
de-installation debugging
disabling, AR:68
enabling, AR:68
facilities, AR:67
de-installing
See also automatic de-installation;
  installing; manual de-installation;
  removing; standard de-installation
OVO agents from managed nodes
  AIX, DCE:54
  automatically, AR:62–AR:63
  HP-UX, DCE:96
  Linux, DCE:140–DCE:141

Master Index

AR  OVO Administrator's Reference
DCE  OVO DCE Agent Concepts and Configuration Guide
CG  OVO Concepts Guide
HTTPS  OVO HTTPS Agent Concepts and Configuration Guide
Master Index

manually, AR:63
MPE/iX, DCE:163
Sequent DYNIX, DCE:230
SGI IRIX, DCE:244
SINIX RM/Reliant, DCE:262
Solaris, DCE:285
Tru64 UNIX, DCE:334
Windows NT/2000, DCE:385
OVPA managed nodes
HP-UX, AR:216
Solaris, AR:216
De-installing Agents, DCE:140
De-installing Agents Automatically, DCE:140
deleing manager responsibilities, CG:468
delete request, HTTPS:161
deleting
message groups, AR:73
node groups, AR:71
delta distribution, HTTPS:94
deny request, HTTPS:161
deploy, HTTPS:31
certificates, HTTPS:169
certificates automatically, HTTPS:158
root certificate, HTTPS:54
deploying suppress unmatched conditions,
CG:356
depot nodes, DCE:86
DESCRIPTION keyword, AR:119
detecting problems
browsing messages effectively,
CG:134–CG:138
eyearly, CG:305
message
event notification, CG:133
severity coloring, CG:139–CG:141
monitoring OVO, CG:131
overview, CG:130
searching object tree, CG:132
viewing messages in message browser,
CG:133
Developer’s Kit APIs man pages, AR:564
DHCP
agent management, HTTPS:154
HTTPS agents, HTTPS:152
NNM synchronization, HTTPS:154
opcnode variables, HTTPS:153
variables, HTTPS:153
Diagnostic Dashboard workspace
accessing OpenView applications, CG:156
overview, CG:83
Diagnostics application, DCE:395
Digital UNIX. See Tru64 UNIX managed
nodes
directories
See also files; target directories; temporary
directories
AIX, DCE:59, DCE:176
HP-UX, DCE:103, DCE:146
maintaining, AR:505
Novell NetWare, DCE:216
runtime data on managed nodes, AR:507
Sequent DYNIX, DCE:233
SGI IRIX, DCE:247
SINIX RM/Reliant, DCE:265
Solaris, DCE:294
Tru64 UNIX, DCE:338
Windows NT/2000, DCE:428
working, AR:463
directory
OVDataDir, HTTPS:36
OVInstallDir, HTTPS:36
structure, HTTPS:36
disabled nodes
See also disabling
description, CG:228
managing, CG:247
disabling
See also disabled nodes; enabling
(de-)installation debugging, AR:68
primary account manually, AR:468
disadvantages of backups
automatic, AR:491
offline, AR:489
Disconnect application, DCE:439
Disk Space application, DCE:173
Disks application, DCE:210
disks, multiple, AR:502–AR:503
Display a File application, DCE:210
display modes, ownership, CG:163,
CG:292–CG:293
display option
ito_op, AR:321
itooprc, AR:323
displaying
available OVO agent versions, AR:65
installed OVO agent versions, AR:65
message
defaults, CG:326
groups, AR:72
dispp<#> pipe file, AR:353
dispq<#> queue file, AR:353
Distributed Computing Environment. See DCE
Distributed Event Interception
configuring, DCE:100
description, DCE:99
distributing
See also distribution
configuration file
other servers, CG:470–CG:471
responsible managers, CG:464
managed nodes
OVO agent configuration, AR:189
scripts and programs, AR:190–AR:194
responsibility in competence centers,
CG:450–CG:451
templates
assigned, CG:315
description, CG:305
message source, CG:315–CG:316
distribution
See also distributing
lists
controlling size, CG:477–CG:479
overview, CG:477–CG:480
manager, AR:191
scripts and programs
AIX, DCE:58–DCE:59
HP-UX, DCE:103–DCE:105
Linux, DCE:144–DCE:146
MPE/iX, DCE:175–DCE:176
Novell NetWare, DCE:215–DCE:216
requirements, AR:190
Sequent DYNIX, DCE:232–DCE:233
SGI IRIX, DCE:246–DCE:247
SINIX RM/Reliant, DCE:264–DCE:265
Solaris, DCE:293–DCE:294
tips, AR:190–AR:193
Tru64 UNIX, DCE:337–DCE:338
UNIX, AR:194
distribution manager, HTTPS:32, HTTPS:92
documentation, related
OVPA, AR:223–AR:224
documenting solutions, CG:40
acknowledging messages, CG:183–CG:184
annotating messages, CG:179–CG:181
overview, CG:178
printing, CG:182
domain, worldwide management, CG:448
Download Configuration Data window
description, AR:486–AR:487
figure, AR:486
opening, AR:487
downloading
collection
data, AR:485–AR:487
files, CG:470
OVPA documentation, AR:223
dragging and dropping nodes, CG:242
See also copying and pasting nodes
dual-homed host, HTTPS:139
duplicate messages
avoiding, CG:417
suppressing
flexible management environments,
CG:378
management server, CG:376–CG:378
overview, CG:370
DYNIX. See Sequent DYNIX managed nodes

E
E message attribute, AR:77
<$E> variable, AR:166
<$e> variable, AR:166
ECS
configuring, DCE:101
elements, preconfigured, AR:71–AR:108
embedded performance component, CG:398
troubleshooting, AR:428–AR:432
EMS
See also EMS Resources application
errors, DCE:119
GUI
closing, DCE:116
overview, DCE:116–DCE:117
starting, DCE:116, DCE:117
viewing resource instances, DCE:116
opcmgr (3) API, DCE:113
overview, DCE:113–DCE:120
## Master Index

<table>
<thead>
<tr>
<th>Term</th>
<th>Page Numbers</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OVO Application Bank window, resource hierarchy</td>
<td>DCE:118–DCE:119</td>
<td></td>
</tr>
<tr>
<td>OVO Application Bank window, command line</td>
<td>DCE:120</td>
<td></td>
</tr>
<tr>
<td>OVO Application Bank window, GUI</td>
<td>DCE:116–DCE:117</td>
<td></td>
</tr>
<tr>
<td>OVO Application Bank window, sending notifications to OVO</td>
<td>DCE:120</td>
<td></td>
</tr>
<tr>
<td>EMS Resources application</td>
<td>See also EMS</td>
<td></td>
</tr>
<tr>
<td>EMS Resources application, description</td>
<td>DCE:118</td>
<td></td>
</tr>
<tr>
<td>EMS Resources application, sample output</td>
<td>DCE:118</td>
<td></td>
</tr>
<tr>
<td>EMS Resources application, syntax</td>
<td>DCE:119</td>
<td></td>
</tr>
<tr>
<td>enabling, See also disabling</td>
<td></td>
<td></td>
</tr>
<tr>
<td>enabling, (de-)installation debugging</td>
<td>AR:68</td>
<td></td>
</tr>
<tr>
<td>enabling, actions on secondary manager</td>
<td>CG:468</td>
<td></td>
</tr>
<tr>
<td>enabling, archive log mode in database</td>
<td>AR:492–AR:493</td>
<td></td>
</tr>
<tr>
<td>enabling, duplicate message suppression on management server</td>
<td>CG:377–CG:378</td>
<td></td>
</tr>
<tr>
<td>enabling, internal OVO error message filtering</td>
<td>AR:384</td>
<td></td>
</tr>
<tr>
<td>operators, to control OVO agents</td>
<td>AR:252–AR:253</td>
<td></td>
</tr>
<tr>
<td>operators, to manage IP networks in IP map</td>
<td>AR:249</td>
<td></td>
</tr>
<tr>
<td>operators, SD-UX</td>
<td>DCE:89</td>
<td></td>
</tr>
<tr>
<td>operators, encapsulator, logfile</td>
<td>CG:384</td>
<td></td>
</tr>
<tr>
<td>Enforced ownership mode</td>
<td>CG:162, CG:294</td>
<td></td>
</tr>
<tr>
<td>Enforced ownership mode, English language</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Enforced ownership mode, English language, agent filesets in OVOPC-CLT</td>
<td>DCE:82</td>
<td></td>
</tr>
<tr>
<td>Enforced ownership mode, English language, components in subproducts</td>
<td>DCE:83</td>
<td></td>
</tr>
<tr>
<td>Enforced ownership mode, English language, character sets</td>
<td>AR:292–AR:294</td>
<td></td>
</tr>
<tr>
<td>Enforced ownership mode, English language, HP-UX configuration and related character sets</td>
<td>AR:298</td>
<td></td>
</tr>
<tr>
<td>Enforced ownership mode, English language, management server</td>
<td>AR:298–AR:301</td>
<td></td>
</tr>
<tr>
<td>Enforced ownership mode, English language, processing managed node files</td>
<td>AR:300–AR:301</td>
<td></td>
</tr>
<tr>
<td>Enforced ownership mode, English language, environmental variables</td>
<td>AR:155</td>
<td></td>
</tr>
<tr>
<td>Enforced ownership mode, English language, environments</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Enforced ownership mode, English language, customizing operator GUI</td>
<td>CG:185</td>
<td></td>
</tr>
<tr>
<td>Enforced ownership mode, English language, character sets</td>
<td>AR:292–AR:294</td>
<td></td>
</tr>
<tr>
<td>description, AR:289</td>
<td></td>
<td></td>
</tr>
<tr>
<td>description, AR:290</td>
<td></td>
<td></td>
</tr>
<tr>
<td>description, CG:187–CG:193</td>
<td></td>
<td></td>
</tr>
<tr>
<td>description, CG:221–CG:224</td>
<td></td>
<td></td>
</tr>
<tr>
<td>description, CG:225–CG:226</td>
<td></td>
<td></td>
</tr>
<tr>
<td>description, AR:289</td>
<td></td>
<td></td>
</tr>
<tr>
<td>errors</td>
<td>EMS: DCE:119</td>
<td></td>
</tr>
<tr>
<td>errors, EMS, getting instructions with opcerr</td>
<td>AR:383</td>
<td></td>
</tr>
<tr>
<td>errors, logfiles, AR:380</td>
<td></td>
<td></td>
</tr>
<tr>
<td>errors, messages</td>
<td></td>
<td></td>
</tr>
<tr>
<td>errors, messages, filtering internal</td>
<td>CG:426, AR:384</td>
<td></td>
</tr>
<tr>
<td>errors, messages, locations</td>
<td>AR:380</td>
<td></td>
</tr>
<tr>
<td>errors, messages, reporting</td>
<td></td>
<td></td>
</tr>
<tr>
<td>errors, messages, reporting, GUI Error Dialog Box</td>
<td>AR:382–AR:383</td>
<td></td>
</tr>
<tr>
<td>errors, messages, reporting, message browser</td>
<td>AR:381</td>
<td></td>
</tr>
<tr>
<td>errors, messages, reporting, overview</td>
<td>AR:380–AR:384</td>
<td></td>
</tr>
<tr>
<td>errors, messages, reporting, stderr and stdout devices</td>
<td>AR:383</td>
<td></td>
</tr>
<tr>
<td>errors, escalating messages</td>
<td>CG:177</td>
<td></td>
</tr>
<tr>
<td>errors, escalating messages, See also messages</td>
<td></td>
<td></td>
</tr>
<tr>
<td>errors, escalating messages, acknowledgements</td>
<td>CG:453</td>
<td></td>
</tr>
<tr>
<td>errors, escalating messages, annotations</td>
<td>CG:454</td>
<td></td>
</tr>
<tr>
<td>errors, escalating messages, guidelines</td>
<td>CG:453</td>
<td></td>
</tr>
<tr>
<td>errors, escalating messages, overview</td>
<td>CG:452–CG:455</td>
<td></td>
</tr>
<tr>
<td>errors, escalating messages, policy</td>
<td>CG:453</td>
<td></td>
</tr>
<tr>
<td>errors, escalating messages, process</td>
<td>CG:454–CG:455</td>
<td></td>
</tr>
<tr>
<td>errors, escalating messages, escmgr template</td>
<td>AR:117</td>
<td></td>
</tr>
<tr>
<td>errors, escalating messages, establishing remote host equivalence</td>
<td>DCE:308</td>
<td></td>
</tr>
<tr>
<td>errors, escalating messages, Ethernet problems</td>
<td>AR:436</td>
<td></td>
</tr>
<tr>
<td>errors, escalating messages, Ethernet Traffic HP, integrating as an OV application</td>
<td>AR:250</td>
<td></td>
</tr>
<tr>
<td>errors, escalating messages, Euro</td>
<td></td>
<td></td>
</tr>
<tr>
<td>errors, escalating messages, Euro, managed node</td>
<td>AR:303</td>
<td></td>
</tr>
<tr>
<td>errors, escalating messages, Euro, management server</td>
<td>AR:305</td>
<td></td>
</tr>
<tr>
<td>errors, escalating messages, Euro, displaying in Motif GUI</td>
<td>AR:278</td>
<td></td>
</tr>
<tr>
<td>errors, escalating messages, Euro, Euro symbol</td>
<td>AR:287</td>
<td></td>
</tr>
</tbody>
</table>
evaluating action results, CG:164
event correlation, CG:435–CG:438
follow-the-sun responsibility switch, AR:148–AR:149
message forwarding between management servers, AR:150–AR:151
responsible managers, CG:463–CG:464
control, AR:502
converting managed node
EUC, AR:303
ROMAN8, AR:300
excluding from automatic backups
database, AR:491

exception warnings, system, AR:343
excluding
files from automatic backups, AR:491
networking commands from streamed jobs, DCE:161
exporting SSP logfiles directory, DCE:308

external
character sets, AR:291–AR:295
monitors, CG:396
nodes
adding, CG:238
characteristics, CG:239

F

<$F> variable, AR:166
Failures policy, CG:134
features
Java and Motif GUIs, AR:320
OVO, CG:17

file tree, management server,
DCE:76–DCE:81
filenames
binary, AR:194
MPE/iX, DCE:171
files
See also directories; include file; logfiles;
makefile
access, CG:226, AR:463
configuration
responsible managers, CG:463–CG:464
control, AR:502
converting managed node
EUC, AR:303
ROMAN8, AR:300
excluding from automatic backups
database, AR:491
temporary, AR:491
HP_OV_consoleSettings, CG:196
include file
  AIX, DCE:63
  HP-UX, DCE:112
  Linux, DCE:151
  MPE/iX, DCE:182
  Novell NetWare, DCE:222
  Sequent DYNIX, DCE:237
  SGI IRIX, DCE:251
  Solaris, DCE:299
  Tru64 UNIX, DCE:351
Windows NT/2000, DCE:432
itooprc, AR:323–AR:327
location
  AIX, DCE:60
  HP-UX, DCE:108
  Linux, DCE:148
  MPE/iX, DCE:177
  Novell NetWare, DCE:217
  Sequent DYNIX, DCE:234
  SGI IRIX, DCE:248
  SINIX RM/Reliant, DCE:266
  Solaris, DCE:295
  Tru64 UNIX, DCE:347
  Windows NT/2000, DCE:430
maintaining, AR:505
makefile
  AIX, DCE:64
  HP-UX, DCE:112
  Linux, DCE:151
  MPE/iX, DCE:183
  Novell NetWare, DCE:223
  Sequent DYNIX, DCE:238
  SGI IRIX, DCE:252
  SINIX RM/Reliant, DCE:270
  Solaris, DCE:300
  Tru64 UNIX, DCE:352
  Windows NT/2000, DCE:432
mapping, DCE:180
.OPC_BRC_History, CG:176
opcinfo, AR:433
OVO agent configuration
  location, AR:362
  types, AR:361
  permissions, AR:463
pipe
  managed nodes, AR:358–AR:359
  management server, AR:353–AR:354
process
  managed node, AR:357–AR:360
  management server, AR:353–AR:354
processing managed node
  English, AR:300–AR:301
processing management server
  ISO 8859-15, AR:299
  Shift JIS, AR:302
queue
  managed nodes, AR:358–AR:359
  management server, AR:353–AR:354
  removing, AR:500
  security, AR:474
SNMP daemon configuration, AR:434
system resource
  AIX, DCE:61
  HP-UX, DCE:109
  MPE/iX, DCE:178
  Novell NetWare, DCE:218
  Sequent DYNIX, DCE:236
  SGI IRIX, DCE:250
  SINIX RM/Reliant, DCE:268
  Solaris, DCE:296
  Tru64 UNIX, DCE:349
  Windows NT/2000, DCE:431
filesets
  list OV installed, HTTPS:186
  basic inventory, HTTPS:186
  detailed inventory, HTTPS:187
  native inventory, HTTPS:187
Filter Messages dialog box
  figure, CG:158
Filter Settings folder
  figure, CG:76
  overview, CG:76–CG:77
filtered message browser
  See also active message browser; history message browser; message browser; pending messages browser
active
  figure, CG:96
  overview, CG:96–CG:97
configuring, CG:209
Master Index

history
figure, CG:98
investigating problems, CG:157–CG:158
overview, CG:98
pending
investigating problems, CG:159
overview, CG:99
saving settings, CG:212–CG:213
filtering messages
conditions, CG:330–CG:354
description, CG:49
internal error messages, CG:426, AR:384
managed node, CG:355
management server, CG:355
multiple templates, CG:328
sources, CG:330–CG:331
Find dialog box
figures
advanced search, CG:132
basic search, CG:132
finding impacted Service Navigator services, CG:156
firewall
internet communication, HTTPS:233
proxies, HTTPS:232
scenarios, HTTPS:232
flexible management
environments
advantages, CG:447
correlating messages, CG:434
overview, CG:446–CG:456
suppressing duplicate messages, CG:378
Japanese-language environments,
AR:305–AR:306
templates
configuring, AR:117–AR:153
examples, AR:146–AR:153
follow-the-sun responsibility switch,
AR:148–AR:149
keywords, AR:119–AR:123
location, AR:117
message forwarding between
management servers, AR:150–AR:151
responsibility switch, AR:146–AR:147
scheduled outages, AR:153
service hours, AR:152
syntax, AR:124–AR:129
types, AR:117
flow charts
communication in competence centers,
CG:451
communication links
central server configuration, CG:462
manufacturing environment, CG:457
configuring
event correlation in OVO, CG:430
message source templates, CG:308
DCE RPC client-server authentication
process, AR:458
directory structure
AIX, DCE:60
HP-UX, DCE:106
Linux, DCE:147
MPE/iX, DCE:177
Novell NetWare, DCE:234
Sequent DYNIX, DCE:234
SGI IRIX, DCE:248
SINIX RM/Reliant, DCE:266
Solaris, DCE:295
Tru64 UNIX, DCE:347
Windows NT/2000, DCE:429
downloading and uploading configuration
files, CG:470
filtering messages
management server, CG:332
multiple templates, CG:328
OVO agent, CG:331
HP-UX configuration and related character
sets
English, AR:298
Japanese, AR:302
installing OVO agents
Novell NetWare, DCE:195
Windows NT/2000, DCE:362
interceptors
MPE/iX console messages, CG:422
SNMP events with NNM, CG:415
logfile encapsulator, CG:384
logical event correlation, CG:428
management responsibility
switching, CG:467
templates for managed nodes, CG:464
message escalation process, CG:454
message flow
  managed nodes, CG:432
  management server, CG:433
message forwarding
  large hierarchies, CG:478
  process, CG:477
OVO
  functional overview, AR:347
  message interface, CG:391
remote actions, AR:472
scalability scenarios
  multiple management servers, CG:489
  multiple management servers with OVO agents and NNM collection stations, CG:491
  NNM collection stations with OVO agents, CG:487
  OVO agents monitoring IP devices, CG:486
  single management server, CG:484
SD-UX remote software depot installation method, DCE:86
SNMP event system in OVO, CG:416
worldwide management domain, CG:448
Flush application, DCE:439
follow-the-sun control, CG:448–CG:450
followthesun template, AR:118
font X resources, AR:279–AR:283
forwmgrp pipe file, AR:353
forwmgrq queue file, AR:353
FTP (re-)installation
See also installing
Windows NT/2000
installing agents, DCE:367–DCE:372
re-installing agents, DCE:378–DCE:381
functionality, OVO, CG:39–CG:43
functions, offline backup, AR:489

G
<$G>$ variable, AR:167
generate certificates, HTTPS:164
generating
  default message
  key relations, CG:366–CG:367
  keys, CG:366–CG:367
  Internet reports, AR:109
  reports, CG:40
generating new NMEV marker,
  DCE:169–DCE:170
generic templates, CG:329
going error instructions
  opcerr, AR:383
  grant request, HTTPS:161
  graphical user interface. See GUI
  group symbols, CG:235
GUI
  See also Java GUI; Motif GUI
documentation
  activating OVO agents
  AIX, DCE:45
  Solaris, DCE:283
  EMS, DCE:116–DCE:117
Java
  accessing, AR:465
  comparison with Motif, AR:318–AR:320
  overview, AR:315–AR:343
  language support
  displaying Euro symbol, AR:278
  font X resources, AR:279–AR:283
  running English GUI in Japanese environment, AR:278
  setting language, AR:277–AR:283
management server, troubleshooting,
  AR:390–AR:392
Motif
  accessing, AR:464
  comparison with Java, AR:318–AR:320
  operator
  saving output, CG:222
  starting OVO, CG:222
Master Index

OVO administrator
accessing, AR:464
description, CG:222
permissions, AR:464–AR:465
SAM, DCE:101
variables, AR:171–AR:186
GUI Error Dialog Box, AR:382–AR:383
guidelines
escalating messages, CG:453
message key, CG:360–CG:363
scripts and programs
notification service, AR:266
trouble ticket system, AR:266

H
HA message group, AR:72
HA resource group, HTTPS:146
HACMP
installation requirements, DCE:48
installing OVO agents, DCE:46–DCE:53
IP
address naming scheme, DCE:47
aliases, DCE:46–DCE:50
troubleshooting, DCE:49
resetting events
HACMP 4.2.2, DCE:51
HACMP 4.3.1, DCE:51–DCE:52
hardware
HP 3000/900, DCE:176
HP 9000/700, DCE:105
HP 9000/800, DCE:105
HP IA64, DCE:105
IBM RS/6000, DCE:59
Intel
Linux, DCE:146
NetWare, DCE:216
Sequent DYNIX, DCE:233
Windows 2000/NT, DCE:428
Siemens Nixdorf, DCE:265
Silicon Graphics, DCE:247
Sun SPARCrstation, DCE:294
Hardware message group
MPE/iX, DCE:165
OVO, AR:72
hardware requirements
OVO
AIX, DCE:33
HP-UX, DCE:69
Linux, DCE:127
MPE/iX, DCE:155
Novell NetWare, DCE:187
Sequent DYNIX, DCE:227
SGI IRIX, DCE:241
SINIX RM/Reliant, DCE:255
Solaris, DCE:273
Tru64 UNIX, DCE:317
Windows NT/2000, DCE:357–DCE:358
headline, message browser
figure, CG:93
heartbeat polling, HTTPS:96
reduce CPU load, HTTPS:96
reduce network load, HTTPS:96
hiding
message browser columns, CG:217
panes and areas, CG:201–CG:203
position controls, CG:198
hie.time.spec template, AR:118
hier.specmgr template, AR:118
hier.time.all template, AR:118
hierarchies
domain, CG:458–CG:459
managed nodes, CG:233–CG:234
management server, CG:457–CG:462
message forwarding, CG:478
hierarchy template, AR:118
hierarchy.agt template, AR:118
hierarchy.sv template, AR:118
history graph
figures
popup menu, CG:155
severity changes over time, CG:154
overview, CG:154
history message browser
See also active message browser; filtered
message browser; message browser;
pending messages browser
investigating problems, CG:157–CG:158
overview, CG:98
hostname, HTTPS:156
automatically changing, HTTPS:135
changing, HTTPS:130
manually changing, HTTPS:130
hostnames
changing, AR:514–AR:526
managed node, AR:522, AR:538
Master Index

hostview application, DCE:306
HP 3000/900, DCE:176
HP 9000/700, DCE:105
HP 9000/800, DCE:105
HP applications, integrating into OVO, AR:245
HP IA64, DCE:105
HP ITO Account
  Windows NT/2000, DCE:364
HP OpenView. See OpenView
HP OpenView Performance Agent. See OVPA
HP OpenView Service Desk, AR:265
HP OpenView VantagePoint Operations. See OVO
HP Software Distributor. See SD-UX
HP System Administrator. See SAM
HP VantagePoint Network Node Manager. See NNM
HP_OV_consoleSettings file, CG:196
hp_ux node group, AR:71
HP-UX 10.x template group, DCE:97
HP-UX 11.x template group, DCE:97
HP-UX managed nodes
  See also HP-UX management server; SD-UX
activating, DCE:93–DCE:96, DCE:331–DCE:333
applications
  ASCII SAM, DCE:101
  EMS Resources, DCE:118–DCE:119
  Motif SAM, DCE:101
EMS
  command line, DCE:120
  GUI, DCE:116–DCE:117
  overview, DCE:113–DCE:120
  OVO Application Bank window, DCE:118–DCE:119
sending notifications to OVO, DCE:120
threshold monitoring, DCE:113–DCE:115
OVO
  accessing programs, AR:465
default operator, DCE:108
de-installing agents, DCE:96
directory structure, DCE:106
file locations, DCE:108
hardware requirements, DCE:69
include file, DCE:112
installation requirements,
  DCE:69–DCE:75
installation tips, DCE:84–DCE:85
installing agents, DCE:84–DCE:92
libraries, DCE:110–DCE:112
logfile locations, AR:508–AR:509
logfile templates, DCE:98
makefiles, DCE:112
manual installation, DCE:90–DCE:92
message templates, DCE:97
organization, DCE:106–DCE:109
overview, DCE:67–DCE:122
preconfigured elements, DCE:97–DCE:102
scripts and programs, DCE:103–DCE:105
SD-UX installation, DCE:86–DCE:92
SNMP event interceptor,
  DCE:99–DCE:101
software requirements, DCE:70–DCE:75
standard installation, DCE:85
system resource files, DCE:109
template groups, DCE:97
OVPA
de-installing, AR:216
installation requirements, AR:210–AR:211
installing, AR:212–AR:215
overview, AR:205–AR:224
preconfigured elements, AR:218–AR:222
template groups, AR:220–AR:222
HP-UX management server
  See also HP-UX managed nodes
configuration and related character sets
  English, AR:298
  Japanese, AR:302
language variable for keyboards, AR:279
HTML format, accessing man pages, AR:557
HTTPS agent
  alternative users, HTTPS:70
  agent profile, HTTPS:77
  changing default port, HTTPS:76
  comparison with DCE agents, HTTPS:84
  configuring the management server, HTTPS:75
  installation, HTTPS:73
  limitations, HTTPS:71
  patching, HTTPS:80
  preparation, HTTPS:72

256
AR  OVO Administrator's Reference
DCE  OVO DCE Agent Concepts and Configuration Guide
CG  OVO Concepts Guide
HTTPS  OVO HTTPS Agent Concepts and Configuration Guide
sudo, HTTPS:81
upgrading, HTTPS:80
architecture, HTTPS:27
authentication troubleshooting, HTTPS:199
certificate troubleshooting, HTTPS:199, HTTPS:208
commands, HTTPS:33
communication troubleshooting,
HTTPS:190, HTTPS:192, HTTPS:204
collect with DCE agent, HTTPS:31
configuration deployment, HTTPS:31
distribution managers, HTTPS:32
multiple parallel configuration servers, HTTPS:32
performance, HTTPS:33
processes, HTTPS:34
resource requirements, HTTPS:32
troubleshooting, HTTPS:35
components, HTTPS:27
configuration deployment, HTTPS:89
configuration push, HTTPS:93
delta distribution, HTTPS:94
directory structure, HTTPS:36
distribution manager, HTTPS:92
firewall and proxies, HTTPS:232
firewall scenarios, HTTPS:232
instrumentation management, HTTPS:90
Internet communication, HTTPS:233
multiple parallel configuration servers, HTTPS:94
network troubleshooting, HTTPS:190
performance, HTTPS:33
processes, HTTPS:34
supported platforms, HTTPS:28
troubleshooting, HTTPS:35

HTTPS agents
DHCP, HTTPS:152
management, HTTPS:154
NNM synchronization, HTTPS:154
opcnodes variables, HTTPS:153
variables, HTTPS:153
heartbeat polling, HTTPS:96
reduce CPU load, HTTPS:96
reduce network load, HTTPS:96
remote control, HTTPS:98

HTTPS communication
advantages, HTTPS:25
commands, HTTPS:37
bbcutil, HTTPS:37
opccsa, HTTPS:39
opccsacm, HTTPS:39
ovc, HTTPS:37
ovcert, HTTPS:39
ovconfch, HTTPS:38
ovconfget, HTTPS:37
ovcoreid, HTTPS:37
ovpolicy, HTTPS:38

HTTPS nodes
add to node bank, HTTPS:162
change hostname
automatically, HTTPS:135
manually, HTTPS:130
change IP address
automatically, HTTPS:135
manually, HTTPS:130
changing hostname, HTTPS:130
changing IP address, HTTPS:130
configuring, HTTPS:100
controlling, HTTPS:88
de-installation
agent software automatically, HTTPS:145
agent software manually, HTTPS:145
problems, HTTPS:145
installation
manual, HTTPS:119
manual behind proxy, HTTPS:143
manually from package files, HTTPS:120
software, HTTPS:101
using clone images, HTTPS:128
map certificate to selected node, HTTPS:163
migrating from DCE, HTTPS:113
migrating to DCE, HTTPS:117
name resolution, HTTPS:136
policy management, HTTPS:90
proxies on management server, HTTPS:144
select all unknown, HTTPS:144
variables, HTTPS:126

I
I message attribute, AR:76
I/O applications, starting remotely, AR:467
IBM AIX. See AIX managed nodes
IBM RS/6000, DCE:59
ICA Browser service, DCE:435
ice_proxy option, AR:323
ice_proxy_address option, AR:324
ice_proxy_advanced option, AR:324
ice_proxy_ftp option, AR:324
ice_proxy_ftp_port option, AR:324
ice_proxy_gopher option, AR:324
ice_proxy_gopher_port option, AR:324
ice_proxy_http option, AR:324
ice_proxy_http_port option, AR:324
ice_proxy_port option, AR:324
ice_proxy_sec option, AR:324
ice_proxy_sec_port option, AR:324
ice_proxy_sock option, AR:325
ice_proxy_sock_port option, AR:325
identifying users logged into Java GUI, AR:343
implementing message policies, CG:303–CG:442
importing
OpenView Operations for Windows policies into OVO, AR:242
OVO templates into OpenView Operations for Windows, AR:241
improving
performance
database, AR:371
Java GUI, AR:342–AR:343
Motif GUI startup, AR:374
OVO, AR:372–AR:373
SNMP management platform, AR:370–AR:371
productivity, CG:305
include file
See also files
AIX, DCE:63
HP-UX, DCE:112
Linux, DCE:151
MPE/iX, DCE:182
Novell NetWare, DCE:222
Sequent DYNIX, DCE:237
SGI IRIX, DCE:251
Solaris, DCE:299
Tru64 UNIX, DCE:351
Windows NT/2000, DCE:432
incoming messages, comparing with match conditions, CG:335–CG:337
Informational ownership mode, CG:163, CG:295
initial node option, AR:322, AR:325
INSERVICE parameter, AR:131
inspecting correlated events in NNM database, CG:431
Install Log application, DCE:425
Install/Update OVO Software and
Configuration window, AR:51, AR:189
install_dir option, AR:325
installation
agent software, HTTPS:101
from clone images, HTTPS:128
key, HTTPS:169
manual, HTTPS:119
manually behind proxy, HTTPS:143
manually from package files, HTTPS:120
OV filesets, HTTPS:186
basic inventory, HTTPS:186
detailed inventory, HTTPS:187
native inventory, HTTPS:187
installation debugging
disabling, AR:68
enabling, AR:68
facilities, AR:67
installation requirements
OVO
AIX, DCE:33–DCE:36
HACMP, DCE:48
HP-UX, DCE:69–DCE:75
Linux, DCE:127–DCE:132
MPE/iX, DCE:155–DCE:156
Novell NetWare, DCE:187–DCE:189
overview, AR:37–AR:40
Sequent DYNIX, DCE:227–DCE:228
SGI IRIX, DCE:241–DCE:242
SINIX RM/Reliant, DCE:255–DCE:256
Solaris, DCE:273–DCE:276
Tru64 UNIX, DCE:317–DCE:320
Windows NT/2000, DCE:357–DCE:360
OVPA
HP-UX, AR:210–AR:211
Solaris, AR:210–AR:211
installation script, AR:48
installation tips
managed nodes
AIX, DCE:37–DCE:39
HP-UX, DCE:84–DCE:85
<table>
<thead>
<tr>
<th>Installation</th>
<th>Platforms</th>
<th>Section</th>
</tr>
</thead>
<tbody>
<tr>
<td>HP-UX, DCE:85–DCE:92</td>
<td></td>
<td>nodes, AR:453</td>
</tr>
<tr>
<td>Linux, DCE:136–DCE:139</td>
<td></td>
<td>servers, AR:452</td>
</tr>
<tr>
<td>MPE/iX, DCE:163</td>
<td></td>
<td>OVO agents on managed nodes, AR:453</td>
</tr>
<tr>
<td>Novell NetWare, DCE:196–DCE:201</td>
<td>overview, AR:35–AR:68</td>
<td>nodes, AR:453</td>
</tr>
<tr>
<td>overview, AR:35–AR:68</td>
<td></td>
<td>servers, AR:452</td>
</tr>
<tr>
<td>SD-UX, DCE:86–DCE:89</td>
<td></td>
<td>OVO agents on managed nodes, AR:453</td>
</tr>
<tr>
<td>Sequent DYNIX, DCE:230</td>
<td></td>
<td>nodes, AR:453</td>
</tr>
<tr>
<td>SGI IRIX, DCE:244</td>
<td></td>
<td>servers, AR:452</td>
</tr>
<tr>
<td>SINIX RM/Reliant, DCE:261</td>
<td></td>
<td>OVO agents on managed nodes, AR:453</td>
</tr>
<tr>
<td>Solaris, DCE:280–DCE:281</td>
<td></td>
<td>nodes, AR:453</td>
</tr>
<tr>
<td>SSH installation method, AR:57–AR:61</td>
<td></td>
<td>servers, AR:452</td>
</tr>
<tr>
<td>Sun Enterprise E10000, DCE:309–DCE:310</td>
<td></td>
<td>OVO agents on managed nodes, AR:453</td>
</tr>
<tr>
<td>Tru64 UNIX, DCE:328</td>
<td></td>
<td>nodes, AR:453</td>
</tr>
<tr>
<td>Tru64 UNIX, DCE:328</td>
<td></td>
<td>servers, AR:452</td>
</tr>
</tbody>
</table>

| OVO managed nodes | OVO configuration on managed nodes, AR:187–AR:203 |
| HP-UX | OVO configuration on managed nodes, AR:187–AR:203 |
| Instant On licenses, AR:510 | OVO configuration on managed nodes, AR:187–AR:203 |
| instruction text interface | OVO configuration on managed nodes, AR:187–AR:203 |
| variables, AR:170 | OVO configuration on managed nodes, AR:187–AR:203 |
| Instructions | OVO configuration on managed nodes, AR:187–AR:203 |
| adding to your message, CG:350 | OVO configuration on managed nodes, AR:187–AR:203 |
| instrumentation management, HTTPS:90 | OVO configuration on managed nodes, AR:187–AR:203 |
| integrated web browser | OVO configuration on managed nodes, AR:187–AR:203 |
| broadcast commands, AR:254 | OVO configuration on managed nodes, AR:187–AR:203 |
| components, AR:245 | OVO configuration on managed nodes, AR:187–AR:203 |
| HP applications, AR:245 | OVO configuration on managed nodes, AR:187–AR:203 |
| HP OpenView plug-in, AR:246 | OVO configuration on managed nodes, AR:187–AR:203 |
| monitoring applications, AR:257 | OVO configuration on managed nodes, AR:187–AR:203 |
| OVO applications, AR:246 | OVO configuration on managed nodes, AR:187–AR:203 |
| Ethernet Traffic HP as OV application, AR:250 | OVO configuration on managed nodes, AR:187–AR:203 |
| IP Activity Monitoring - Tables as OV service, AR:251 | OVO configuration on managed nodes, AR:187–AR:203 |
| monitoring programs, CG:394 | OVO configuration on managed nodes, AR:187–AR:203 |
| SMS into OVO, DCE:442–DCE:447 | OVO configuration on managed nodes, AR:187–AR:203 |
| Sun Management Center, DCE:311 | OVO configuration on managed nodes, AR:187–AR:203 |
| Intel | OVO configuration on managed nodes, AR:187–AR:203 |
| Linux, DCE:146 | OVO configuration on managed nodes, AR:187–AR:203 |
| NetWare, DCE:216 | OVO configuration on managed nodes, AR:187–AR:203 |
| Sequent DYNIX, DCE:233 | OVO configuration on managed nodes, AR:187–AR:203 |
| intercepting events | OVO configuration on managed nodes, AR:187–AR:203 |
| ECS, DCE:101 | OVO configuration on managed nodes, AR:187–AR:203 |
messages
- applications, AR:259
- description, CG:37
- managed nodes, CG:37
- MPE/iX console, CG:422–CG:423
- MPE/iX managed nodes,
  DCE:165–DCE:170
SNMP
- events, CG:414–CG:415
- traps, CG:414
interceptor, event, CG:431
interface, message, CG:391–CG:392
internal nodes
- adding, CG:236
- characteristics, CG:237
Internet reports, generating, AR:109
interoperability
- overview, AR:225–AR:242
- OVO and OpenView Operations for
  Windows, AR:227–AR:242
interval, refresh, CG:193
intervals, setting time, CG:466
investigating problems
- accessing OpenView applications, CG:156
- examining message attributes, CG:144
- finding impacted Service Navigator
  services, CG:156
message
- browser, CG:143
- histories, CG:157–CG:158
- modifying message attributes, CG:145
- overview, CG:142–CG:143
- pending messages browser, CG:159
- reviewing original message text, CG:146
- viewing
  - custom message attributes,
    CG:147–CG:148
  - message severity, CG:151–CG:155
- workspace pane, CG:150
IP
- addresses
  - changing, AR:514–AR:526
  - managed node, AR:522, AR:538
  - management server, AR:515–AR:517,
    AR:527–AR:530
- devices, CG:486
HACMP
- address naming scheme, DCE:47
- aliases, DCE:46–DCE:50
- troubleshooting, DCE:49
map
- accessing with Jovw, AR:336–AR:338
- network management, AR:249
- submaps, CG:241
- troubleshooting point-to-point and
  Ethernet problems, AR:436
IP Activity Monitoring - Tables, integrating
  as OV service, AR:251
IP address, HTTPS:156
- automatically changing, HTTPS:135
- changing, HTTPS:130
- manually changing, HTTPS:130
IRIX. See SGI IRIX managed nodes
ISO 8859-15
- on managed nodes, AR:287
- on management server, AR:299
- ito_op startup script, AR:321–AR:322
- ito_restore.sh script, AR:497
itop, CG:60
- See also opc_op; netop
J
Japanese language
- character sets, AR:294
- flexible management, AR:305–AR:306
- HP-UX configuration and related character
  sets, AR:302
- management server, AR:302–AR:304
- processing managed node files,
  AR:303–AR:304
Java GUI
- See also GUI; Motif GUI documentation
- accessing
  - Jovw, AR:336–AR:338
  - NNM, AR:328–AR:330
  - OVO, AR:465
- accessing quick filters, CG:214
- adding tabs to browser pane, CG:214
- applications, AR:174
- browser pane, CG:89–CG:91
- changing
  - look and feel, CG:197
  - operator passwords, CG:186
refresh interval, CG:193
choosing web browser, CG:204
comparison with Motif GUI, AR:318–AR:320
configuring filtered message browsers,
   CG:209
customizing
   message browser columns, CG:216
   message event notification, CG:208
   overview, CG:185
   popup menus, CG:206–CG:207
   shortcut bar, CG:204
figure, CG:65
hiding
   message browser columns, CG:217
   panes and areas, CG:201–CG:203
   position controls, CG:198
   identifying logged-in users, AR:343
   ito_op startup script, AR:321–AR:322
   itooprc file, AR:323–AR:327
loading default configuration,
   CG:187–CG:193
menu bar, CG:106
moving panes and areas, CG:199
object pane, CG:69–CG:70
OpenView applications, AR:330–AR:332
overview, AR:315–AR:343
performance tips, AR:342–AR:343
popup menus, CG:110
position controls, CG:109
saving
   console settings, CG:195–CG:197
   message browser filter, CG:212–CG:213
   message browser layout, CG:218
   shortcut bar, CG:67–CG:68
   showing
   message browser columns, CG:217
   panes and areas, CG:201–CG:203
   position controls, CG:198
   startup options, AR:321–AR:322
   status bar, CG:104
switching message colors to entire line,
   CG:215
toolbar, CG:107
tour, CG:65–CG:66
variables, AR:171–AR:186
web browsers, CG:100–CG:103
workspace pane, CG:79–CG:81
Job message group
   MPE/iX, DCE:165
   OVO, AR:72
Job Status application, DCE:400
Jovw
   accessing, AR:336–AR:338
   default IP map, AR:336–AR:338
Just-in-Time compiler. See JVM JIT compiler
K
kernel parameters, AR:38
key store, HTTPS:48
keyboards, setting language variable on HP-UX, AR:279
keys, message, CG:365
keywords, template
   flexible management, AR:119–AR:123
time, AR:144–AR:145
L
Lan Console application, DCE:173
language support
   GUI
      displaying Euro symbol, AR:278
      font X resources, AR:279–AR:283
      running English GUI in Japanese environment, AR:278
      setting language, AR:277–AR:283
managed nodes
   managing English nodes with Japanese management server, AR:291
   overview, AR:284–AR:297
   setting character set, AR:287
   setting language, AR:286
management server
   overview, AR:275–AR:283
   setting character set, AR:276
   setting language, AR:275
languages
   OVO
      other, AR:312
libraries
   AIX, DCE:62–DCE:64
   HP-UX, DCE:110–DCE:112
   Linux, DCE:150–DCE:151
Master Index

managed nodes, AR:544
MPE/iX, DCE:182–DCE:183
Novell NetWare, DCE:222–DCE:223
Sequent DYNIX, DCE:237–DCE:238
SGI IRIX, DCE:251–DCE:252
SINIX RM/Reliant, DCE:269–DCE:270
Solaris, DCE:298–DCE:300
Tru64 UNIX, DCE:350–DCE:352
Windows NT/2000, DCE:432
Licence Overview, AR:112
License application, DCE:439
licenses
command-line tool, AR:512–AR:513
Instant On, AR:510
maintaining, AR:510–AR:513
types, AR:510–AR:511
linking messages logically, CG:46
Linux (RedHat) template group, DCE:142
Linux managed nodes
default operator, DCE:148–DCE:149
de-installing agents, DCE:140
directory structure, DCE:147
file locations, DCE:148
hardware requirements, DCE:127
include file, DCE:151
installation
requirements, DCE:127–DCE:132
tips, DCE:135–DCE:136
installing agents, DCE:136–DCE:139
libraries, DCE:150–DCE:151
logfile templates, DCE:142
makefile, DCE:151
organization, DCE:147–DCE:149
overview, DCE:125–DCE:152
preconfigured elements, DCE:142–DCE:143
removing agents, DCE:141
scripts and programs, DCE:144–DCE:146
SNMP event interceptor (not supported),
DCE:143
software requirements, DCE:128–DCE:132
template groups, DCE:142
List Processes application, AR:218
List Versions application, AR:218
lists, message distribution, CG:477–CG:480
LM Sessions application, DCE:401
Load/Unload an arbitrary NLM application,
DCE:211
loading default configuration, CG:187–CG:193
Local Location Broker
troubleshooting, AR:427
Local Users application, DCE:402
LOCAL_ON_JAVA_CLIENT variable,
AR:170
LOCAL_ON_JAVA_CLIENT_WEB variable,
AR:170
locale option, AR:322, AR:325
localizing object names, AR:313
locating
See also location
messages, CG:317
location
See also locating
configuration data, AR:485
error messages, AR:380
files
AIX, DCE:60
HP-UX, DCE:108
Linux, DCE:148
managed node logfiles, AR:508–AR:509
managed node processes, AR:360
MPE/iX, DCE:177
Novell NetWare, DCE:217
opcinfo on managed nodes, AR:377
OVO agent configuration, AR:377
Sequent DYNIX, DCE:234
SGI IRIX, DCE:248
SINIX RM/Reliant, DCE:266
Solaris, DCE:295
Tru64 UNIX, DCE:347
Windows NT/2000, DCE:430
scripts and programs
AIX, DCE:58
HP-UX, DCE:103
Linux, DCE:145
MPE/iX, DCE:175
Novell NetWare, DCE:217
Sequent DYNIX, DCE:233
SGI IRIX, DCE:246
SINIX RM/Reliant, DCE:266
Solaris, DCE:293
Tru64 UNIX, DCE:337
Windows NT/2000, DCE:427
templates
flexible management, AR:117
message forwarding, AR:137
scheduled outage, AR:130
scheduled outages, AR:130
service hours, AR:130
${LOGFILE} variable, AR:162
logfile
See also files
application, monitoring, AR:258
capsulator
changing character set, AR:291
character sets supported, AR:295–AR:297
description, CG:384
flow chart, CG:384
error messages, AR:380
locations on managed nodes,
AR:508–AR:509
messages, CG:384–CG:390
SSP directory, exporting, DCE:308
templates
Citrix MetaFrame, DCE:437
defining, CG:388
description, CG:385
HP-UX (OVO), DCE:98
Linux, DCE:142
SGI IRIX, DCE:245
Solaris (OVO), DCE:288
Sun Enterprise E10000, DCE:304
Tru64 UNIX, DCE:335
variables, AR:162
logging, HTTPS:189
logging data with OVPA, AR:208
logging messages, CG:37, CG:379–CG:380
login
DCE, AR:468
RPC, AR:457
Logon Report, AR:110
LOGONLY parameter, AR:131
${LOGPATH} variable, AR:162
logs, redo, AR:503

M
magmgrp pipe file, AR:353
magmgrp queue file, AR:353
maintaining
database, AR:501
directories, AR:505
files, AR:505
licenses, AR:510–AR:513
managed nodes, AR:506–AR:509
OpenView, AR:504
Major message severity level, AR:74
makefile
See also files
AIX, DCE:64
HP-UX, DCE:112
Linux, DCE:151
MPE/iX, DCE:183
Novell NetWare, DCE:223
Sequent DYNIX, DCE:238
SGI IRIX, DCE:252
SINIX RM/Reliant, DCE:270
Solaris, DCE:300
Tru64 UNIX, DCE:352
Windows NT/2000, DCE:432
man pages
accessing
command line, AR:557
HTML format, AR:557
APIs
Developer's Kit, AR:564
OVO, AR:562
OVO, AR:555–AR:564
printing, AR:557
Service Navigator, AR:563
managed nodes
See also Managed Nodes window;
management server
accessing MIB, AR:433–AR:434
adding to OVO
description, CG:229
from IP submaps, CG:241
from OVO Add Node window,
CG:242–CG:245
in Node Bank window, AR:49
overview, CG:236–CG:248
with templates, CG:314
APIs, AR:543
building, CG:227
character sets
changing, AR:290
EUC, AR:303
external, AR:291–AR:295
ROMAN8, AR:300
Master Index

Shift JIS, AR:306
communication types, AR:54–AR:56
configuring
  authenticated RPCs, AR:454
  DCE cell, AR:454
description, CG:38
hierarchies, CG:459
  regional management servers,
    CG:461–CG:462
copying and pasting, CG:242
  correlating messages, CG:429, CG:432
debugging software (de-)installation,
  AR:67–AR:68
defaults, CG:246
de-installing OVO agents
  automatically, AR:62–AR:63
  manually, AR:63
description, CG:37–CG:38
directories with runtime data, AR:507
disabled, CG:247
distributing
  OVO agent configuration, AR:189
  scripts and programs, AR:190–AR:194
dragging and dropping, CG:242
external
  adding, CG:238
  characteristics, CG:239
files
  pipe, AR:358–AR:359
  process, AR:358–AR:359
  queue, AR:358–AR:359
filtering messages, CG:355
group symbols, CG:235
hostnames and IP addresses, AR:522, AR:538
installing
  OVO agents, AR:35–AR:68
  OVO configuration, AR:187–AR:203
internal
  adding, CG:236
  characteristics, CG:237
kernel parameters, AR:38
language support, AR:284–AR:297
libraries, AR:344
logfile locations
  AIX, AR:508
  HP-UX, AR:509
  HP-UX 10.x/11.x, AR:508
  MPE/iX, AR:508
  OVO, AR:508–AR:509
  Solaris, AR:509
  Windows NT, AR:508
maintaining, AR:506–AR:509
managing OVO agents, AR:64–AR:66
message-allowed, CG:228
multiple parent groups, CG:235
opcinfo file, AR:377
operating systems
  AIX, DCE:31–DCE:65
  HP-UX, DCE:67–DCE:122
  Linux, DCE:125–DCE:152
  MPE/iX, DCE:153–DCE:183
  Novell NetWare, DCE:185–DCE:223
  Sequent DYNIX, DCE:225–DCE:238
  SGI IRIX, DCE:239–DCE:252
  SNIX RM/Reliant, DCE:253–DCE:270
  Solaris, DCE:271–DCE:313
  Tru64 UNIX, DCE:315–DCE:353
organizing, CG:227–CG:250
passwords
  assigning, AR:468–AR:470
  DCE, AR:467–AR:468
  MPE/iX, AR:469
  Novell NetWare, AR:470
  UNIX, AR:469
  Windows NT, AR:470
process files, AR:357–AR:360
processes, AR:355–AR:362
processing files
  English, AR:300–AR:301
  redistributing scripts, AR:488
returning names with pattern matching,
  AR:334
security, CG:247
starting
  applications, AR:261–AR:262
  broadcast commands, AR:261–AR:262
  templates for responsible managers, CG:464
  troubleshooting
    all managed nodes, AR:401–AR:415
embedded performance component, AR:428–AR:432
mixed-case node names, AR:394
UNIX, AR:393, AR:416–AR:419
Windows, AR:399–AR:400
types, CG:228
updating
OVO agents, AR:48–AR:56
OVO configuration, AR:187–AR:203
windows, CG:228
Managed Nodes window
description, CG:60
management hierarchies
See also management server
overview, CG:457–CG:462
profiles, CG:457
responsibilities, CG:458–CG:459
setup ratio, CG:458
management profiles, CG:457
See also management server
management responsibility
See also management server
domain hierarchies, CG:458–CG:459
message forwarding between management servers, AR:150–AR:151
switch, AR:146–AR:147
follow-the-sun, AR:148–AR:149
template syntax, AR:126
management server
See also managed nodes; management hierarchies; management profiles; management responsibility; managers
action-allowed
configuring, CG:459
specifying, CG:469
backing up data, AR:488–AR:500
central
configuring, CG:462
description, CG:459
changing hostnames or IP addresses,
competence centers, CG:450–CG:451
configuring
English language, AR:298–AR:301
OpenView Operations for Windows to forward messages to OVO,
AR:235–AR:240
OVO agents for OpenView Operations for Windows, AR:228–AR:231
connecting to trouble ticket systems,
CG:480
converting to EUC, AR:305
correlating messages, CG:429, CG:433
default setup, CG:446
description, CG:36
distributing configuration, CG:470–CG:471
duplicate messages
enabling suppression, CG:377–CG:378
suppressing, CG:376
escalating messages, CG:452–CG:455
files
pipe, AR:353–AR:354
process, AR:353–AR:354
queue, AR:353–AR:354
filtering messages, CG:355
flexible architecture, CG:447
follow-the-sun control, CG:448–CG:450
forwarding messages
between management servers,
CG:472–CG:483
OpenView Operations for Windows,
AR:236
hierarchies, CG:457–CG:462
installation tips, AR:45
language support
overview, AR:275–AR:283
setting character set, AR:276
setting language, AR:275
multiple, CG:443–CG:491
OVO file tree, DCE:76–DCE:81
primary, CG:446
processes, AR:349–AR:354
processing files
ISO 8859-15, AR:299
Shift JIS, AR:302
processing messages, CG:332
reconfiguring after changing hostname or IP address, AR:518–AR:522, AR:531–AR:537
regional
   configuring, CG:461–CG:462
description, CG:458
responsibility
   configuring, CG:463–CG:471
   switching, CG:467–CG:469
secondary, CG:460
sending messages
   OpenView Operations for Windows, AR:228
   OVO, AR:232
single, CG:484
software sub-tree
customer-specific, DCE:81
vendor-specific, DCE:80
troubleshooting
   GUI, AR:390–AR:392
   server, AR:388–AR:389
management, flexible, CG:446–CG:456
managers
   See also management server
   action-allowed
      adding, CG:469
central server, CG:459
backup, CG:469
primary
   changing, CG:467–CG:469
   initial, CG:446
responsibility, CG:463–CG:471
secondary, CG:460
managing
   disabled nodes, CG:247
message source templates, CG:307–CG:316
messages, CG:49
OVO agents, AR:64–AR:66
Sun Enterprise E10000, DCE:301–DCE:302
managing certificates, HTTPS:161
manual de-installation
   See also de-installing
OVO
   AIX, DCE:54
   HP-UX, DCE:96
   Linux, DCE:140
SINIX RM/Reliant, DCE:262
Solaris, DCE:285
Tru64 UNIX, DCE:334
Windows NT/2000, DCE:385
OVPA
   HP-UX, AR:217
   Solaris, AR:217
manual installation
   See also installing
   instrumentation, HTTPS:91
OVO
   AIX, DCE:42–DCE:45
   HP-UX, DCE:90–DCE:92
   Linux, DCE:137–DCE:139
   SINIX RM/Reliant, DCE:261
   Solaris, DCE:280
   Windows NT/2000, DCE:382–DCE:384
OVPA
   HP-UX, AR:213
   Solaris, AR:213
   policies, HTTPS:91
manufacturing environment
   communication links, CG:457
   management profiles, CG:457
mapped requests
   select all, HTTPS:162
mapped to, HTTPS:156
mapping
   ARPA hostnames to NS node names
      overview, DCE:178–DCE:181
      problems, DCE:180
      resolving names, DCE:181
   vt3k operation, DCE:179
   MPE/iX messages to OVO security levels,
      DCE:166
   NMEV markers, DCE:166–DCE:169
   marking messages, CG:292
   match conditions, comparing with incoming
      messages, CG:335–CG:337
   mathematical operators in
      pattern-matching, CG:338–CG:339
   max_limited_messages option, AR:322,
      AR:325
   maximum threshold, CG:401
MC/ServiceGuard
   support, DCE:121
   Memory Load application, DCE:403
   Memory Use application, DCE:211

266

AR  OVO Administrator's Reference
DCE  OVO DCE Agent Concepts and Configuration Guide
CG  OVO Concepts Guide
HTTPS  OVO HTTPS Agent Concepts and Configuration Guide
### Master Index

| menu bar | switching colors to entire line, CG:215 |
| figure, CG:106 | viewing  |
| overview, CG:106 | custom message attributes, CG:148 |
| merging multiple certificate servers | messages, CG:133 |
| environments, HTTPS:56 | Message Browser window |
| message | description, CG:61 |
| defaults | message attributes and values, AR:73 |
| message correlation options, CG:325 | overview, AR:73–AR:77 |
| output options for a message stream | Message Condition Advanced Options |
| interface, CG:325 | window, CG:418 |
| pattern-matching options, CG:325 | message conditions |
| message-allowed managed nodes, CG:228 | See also messages |
| Message and Suppress Conditions window, CG:337 | defining advanced options, CG:408 |
| message attributes | setting up, CG:333–CG:334 |
| setting defaults, CG:324 | message correlation options |
| message browser | setting defaults, CG:325 |
| See also active message browser; filtered | Message Correlation window, CG:360 |
| message browser; history message | Message Dashboard workspace |
| browser; pending messages browser | current state chart, CG:152 |
| accessing quick filters, CG:214 | history chart, CG:154 |
| browsing effectively, CG:134–CG:138 | overview, CG:82 |
| configuring filters | viewing message severity, CG:151–CG:155 |
| active, CG:96–CG:97 | message event notification |
| history, CG:98 | customizing, CG:208 |
| overview, CG:209 | overview, CG:133 |
| pending, CG:99 | message event warning, CG:133 |
| consolidating messages, CG:306 | Message Group Bank window, AR:72 |
| customizing columns | message groups |
| message attributes, CG:136 | See also Message Groups window; messages |
| physical layout, CG:216 | adding, AR:73 |
| figures | adding new, CG:252 |
| browser pane, CG:90 | default, AR:71–AR:77 |
| custom message attributes, CG:148 | defining, CG:50 |
| workspace pane, CG:91 | deleting, AR:73 |
| hiding columns, CG:217 | displaying, AR:72 |
| investigating problems, CG:143 | modifying, AR:73 |
| Java and Motif GUIs, AR:318 | organizing, CG:251–CG:252 |
| operator, CG:223 | reviewing, CG:252 |
| overview, CG:92–CG:93 | Message Groups folder |
| OVO administrator, CG:223–CG:224 | colors, CG:73 |
| reporting errors, AR:381 | figure, CG:73 |
| reusing filters, CG:212–CG:213 | organizing, CG:74 |
| saving | overview, CG:73–CG:74 |
| customized layout, CG:218 | Message Groups window, CG:60 |
| filter to object pane, CG:214 | See also message groups |
| showing columns, CG:217 | message keys, CG:359 |
| See also messages | default, CG:366–CG:367 |

**AR**  
OVO Administrator’s Reference

**DCE**  
OVO DCE Agent Concepts and Configuration Guide

**CG**  
OVO Concepts Guide

**HTTPS**  
OVO HTTPS Agent Concepts and Configuration Guide
Master Index

guidelines, CG:360–CG:363
relations, CG:366–CG:367
message operations template syntax, AR:127
Message Properties dialog box
figures
Annotations tab, CG:180
Custom Attributes tab, CG:149
General tab, CG:95
Instructions tab, CG:168
Original Message tab, CG:146
message settings
assigning, CG:347
message source templates
See also Message Source Templates
window; message sources; messages
configuring, CG:308
creating, CG:309
distributing, CG:315–CG:316
elements, CG:307
managing, CG:307–CG:316
variables, AR:155–AR:169
Message Source Templates window
See also message source templates
description, CG:309
figure, CG:316
Templates Groups list box, CG:310
message sources
See also message source templates; messages
evaluating, CG:317–CG:318
filtering, CG:330–CG:331
message stream interface output options
setting defaults, CG:325
Message Stream Interface. See MSI
message target rules template syntax,
AR:127
message_notification_dlg option, AR:325
message_notification_dlg_app option, AR:325
message_notification_dlg_app_path option,
AR:325
message_notification_show_all option,
AR:325
messages
See also acknowledgements;
acknowledging; escalating messages;
message browser; message conditions;
message groups; message keys;
message source templates; message
sources
acknowledging
automatically, CG:166
overview, CG:183–CG:184
with message keys, CG:365
annotating, CG:179–CG:181
annotating acknowledged, CG:366
API, CG:391–CG:392
attributes, AR:75–AR:77
resolving, CG:323
time, CG:449
browsing effectively, CG:134–CG:138
buffering, CG:37, CG:439
parameters, AR:132
catalogue, CG:318
classifying unmatched, CG:49
closing, CG:178
collecting, CG:319–CG:321
colors
overview, CG:94
switching, CG:215
comparing, CG:37
conditions, specifying, CG:390
consolidating in browser, CG:306
control-switched, CG:473
correcting, CG:393
correlating, CG:359
different sources, CG:429
flexible management environments,
CG:434
managed nodes, CG:432
management server, CG:433
types, CG:359
with events, CG:357
customizing columns, CG:136
defaults, CG:324–CG:325, CG:326
custom message attributes, CG:324
message attributes, CG:324
details, CG:144
escalated message, CG:452
distribution lists, CG:477–CG:480
duplicate
SNMP devices, CG:417
error, AR:380
Master Index

escalating, CG:177, CG:452–CG:455

evaluating
severity, CG:318
examining attributes, CG:144

filtering, CG:49
managed node, CG:355
management server, CG:355
sources, CG:330–CG:331
strategies, CG:355–CG:378
through multiple templates, CG:328
with conditions, CG:330–CG:354

formatting, CG:50
forwarding, CG:449
between management servers,
notification system, AR:133
OpenView Operations for Windows
management server, AR:236
strategies, CG:480–CG:482
template, AR:137–AR:139
trouble ticket system, AR:133
unmatched messages, AR:382
generating
continuous, CG:405
policy, CG:402–CG:405
with reset, CG:403
without reset, CG:404
groups, CG:50
incoming, CG:335–CG:337

intercepting
application messages, AR:259
description, CG:37
MPE/iX managed nodes,
DCE:165–DCE:170
interface, CG:391–CG:392

investigating
message histories, CG:157–CG:158
pending messages, CG:159
keys, CG:359
linking logically, CG:46
locating, CG:317
logfile, CG:384–CG:390
logging
description, CG:37
results, CG:379–CG:380

managing, CG:49, CG:305–CG:306
marking, CG:292
modifying attributes, CG:145
MPE/iX console
overview, CG:422–CG:425
variables, AR:164
notification, CG:475–CG:476
overview, CG:45–CG:50, CG:95
owning, CG:162–CG:163, CG:292,
CG:292–CG:295
pattern-matching, CG:338–CG:346
processing
description, CG:46–CG:48
on management server, CG:332
overview, CG:322–CG:329
quantity, reducing, CG:357–CG:378
regrouping, CG:312, CG:381–CG:383
reset, sending automatically,
CG:367–CG:369
responding, CG:50
reviewing
details, CG:95
original text, CG:146
scanning, CG:134
scheduled action variables, AR:169
sending to management server
OpenView Operations for Windows,
AR:228
OVO, AR:232
severity
coloring, CG:139–CG:141
viewing in Message Dashboard,
CG:151–CG:155
severity levels, AR:74–AR:75
status, CG:319
suppressing
duplicate, CG:370
multiple, CG:329
switching control, CG:473–CG:474
target rules, CG:465–CG:466
template conditions, CG:46
templates, CG:389
threshold monitors, CG:393–CG:413
unbuffering, CG:39
automatically, CG:439
manually, CG:439–CG:440
viewing
  in message browser, CG:133
metrics See performance metrics
MF_ICA_Browser object, DCE:436
MF_Prog_Neighbourhood object, DCE:436
MIB
  managed node, AR:433–AR:434
  object monitors, CG:395
Microsoft. See Windows NT/2000 managed nodes
midaemon monitor template, AR:221
minimum threshold, CG:401
Minor message severity level, AR:74
Mirrored Devices application, DCE:211
mirrored online redo logs, AR:503
Misc message group
  MPE/iX, DCE:165
  OVO, AR:72
missing OS patches for Solaris, DCE:279
mixed clusters, AR:194
moa* temporary file, AR:359
modes
  archive log
    database, AR:488, AR:491
    enabling, AR:492–AR:493
    auditing, AR:475
    ownership, CG:162, CG:293–CG:295
    ownership display, CG:163, CG:292–CG:293
  MPE/iX, DCE:436
  OVO, AR:72
  managed node, AR:433–AR:434
  object monitors, CG:395
Microsoft. See Windows NT/2000 managed nodes
midaemon monitor template, AR:221
minimum threshold, CG:401
Minor message severity level, AR:74
Mirrored Devices application, DCE:211
mirrored online redo logs, AR:503
Misc message group
  MPE/iX, DCE:165
  OVO, AR:72
missing OS patches for Solaris, DCE:279
mixed clusters, AR:194
moa* temporary file, AR:359
modes
  archive log
    database, AR:488, AR:491
    enabling, AR:492–AR:493
    auditing, AR:475
    ownership, CG:162, CG:293–CG:295
    ownership display, CG:163, CG:292–CG:293
Modify Message Attributes dialog box
  figure, CG:145
Modify OVO Interface Messages window,
  CG:392
modifying
  conditions, CG:338
  logfile templates on Tru64 UNIX, DCE:335
  message groups, AR:73
  node groups, AR:71
MoM
  merging, HTTPS:56
  sharing a certificate server, HTTPS:61
monagtq queue file, AR:358
monitor agent, CG:395–CG:400
See also monitoring
Monitor Console application, DCE:173
monitored objects
See also monitoring
Citrix MetaFrame, DCE:436
MPE/iX, DCE:171
Sun Enterprise E10000, DCE:305
monitoring
See also monitor agent; monitored objects
application
  integration, AR:257
  logfiles, AR:258
environment, CG:131
managed nodes, CG:228
objects
  external, CG:397
  MIB, CG:396
  program, CG:396
performance metrics, CG:398
performance with NMA, DCE:206
programs, CG:394
SMS, DCE:445
Sun Enterprise E10000, DCE:301–DCE:302
variables, CG:401
Motif GUI
  accessing, AR:464
  comparison with Java GUI, AR:318–AR:320
  improving performance, AR:374
  variables, AR:171–AR:186
Motif GUI documentation
  See also GUI; Java GUI
Motif SAM, DCE:101
moving
  panes and areas, CG:199
MPE/iX console
  See also MPE/iX managed nodes
  accessing programs, AR:465
messages
  advanced options, CG:424
  condition examples, CG:424–CG:425
  intercepting, CG:422–CG:423
  interceptor, CG:422
  overview, CG:422–CG:425
  templates, CG:423–CG:424
  variables, AR:164
MPE/iX managed nodes
  See also MPE/iX console
  agent jobs, DCE:159
  applications, DCE:172–DCE:174
  DCE daemon, DCE:157
  default operator, DCE:158, DCE:177
de-installing agents, DCE:163
directory structure, DCE:177
domain name resolution, DCE:159
executable libraries, DCE:159
file locations, DCE:177
filename tips, DCE:171
hardware requirements, DCE:155
include file, DCE:182
installation
requirements, DCE:155–DCE:156
tips, DCE:157–DCE:160
installing agents, DCE:163
intercepting messages
default message mapping, DCE:165
generating new NMEV marker,
DCE:169–DCE:170
mappings messages to OVO security levels,
DCE:166
mapping NMEV markers,
DCE:166–DCE:169
overview, DCE:165–DCE:170
IP addresses, DCE:158
languages, DCE:158
libraries, DCE:182–DCE:183
logfile
locations, AR:508
logging group, DCE:159
login and logout UDCs, DCE:158
makefile, DCE:183
mapping ARPA hostnames to NS node names
overview, DCE:178–DCE:181
problems, DCE:180
resolving names, DCE:181
vt3k operation, DCE:179
monitored objects, DCE:171
NCS daemon, DCE:157
organization, DCE:177–DCE:181
overview, DCE:153–DCE:183
passwords, AR:469
preconfigured elements, DCE:164–DCE:174
scripts and programs, DCE:175–DCE:176
SNMP event interceptor (not supported),
DCE:171
software requirements, DCE:155–DCE:156
spool files, DCE:160
streamed jobs
customizing job stream facility, DCE:162
excluding networking commands,
DCE:161
overview, DCE:161–DCE:163
preparing OVO, DCE:163
starting, DCE:161
SYSSTART.PUB.SYS parameters,
DCE:161
system resource file, DCE:178
time zones, DCE:160
troubleshooting
installation, AR:395–AR:398
runtime, AR:420–AR:426
mpicdmp pipe file, AR:353
mpicdmq queue file, AR:353
mpicmap pipe file, AR:358
mpicmaq queue file, AR:358
mpicummp pipe file, AR:353
mpicummq queue file, AR:353, AR:354
mpimap pipe file, AR:358
mpimaq queue file, AR:358
mpimmp pipe file, AR:354
<$MSG_APPL> variable, AR:155
<$MSG_GEN_NODE> variable, AR:156
<$MSG_GEN_NODE_NAME> variable,
AR:156
<$MSG_GRP> variable, AR:156
<$MSG_ID> variable, AR:156
<$MSG_NODE> variable, AR:156
<$MSG_NODE_ID> variable, AR:157
<$MSG_NODE_NAME> variable, AR:157
<$MSG_OBJECT> variable, AR:157
<$MSG_SEV> variable, AR:157
<$MSG_TEXT> variable, AR:158
<$MSG_TIME_CREATED> variable, AR:158
<$MSG_TYPE> variable, AR:158
msgagtdf file, AR:358
msgagtp pipe file, AR:358
msgagtq queue file, AR:358
msgforw template, AR:119
MsgGroup message attribute, AR:77
msgip pipe file, AR:358
msgiq queue file, AR:358
msgmgrp pipe file, AR:354
msgmgrq queue file, AR:354
msgmni parameter, AR:38
MSGTARGETMANAGERS keyword, AR:121
MSGTARGETRULES keyword, AR:122
MSGTARGETRULECONDS keyword,
AR:122
AR OVO Administrator’s Reference
DCE OVO DCE Agent Concepts and Configuration Guide
CG OVO Concepts Guide
HTTPS OVO HTTPS Agent Concepts and Configuration Guide
multi-homed host, HTTPS:139
multi-homed hosts, troubleshooting, AR:435–AR:442
multiple
disks for configuring database, AR:502–AR:503
management servers, CG:443–CG:491
messages, suppressing, CG:329
operators, CG:55
parent groups, CG:235
templates
configuring, CG:326
processing simultaneously,
CG:327–CG:328
multiple certificate servers, HTTPS:55, HTTPS:59
multiple parallel configuration servers, HTTPS:94

N
N message attribute, AR:76
$$N$$ variable, AR:167
$$NAME$$ variable, AR:163
name resolution, HTTPS:136
navigating template group hierarchies, CG:311
NCP Info application, DCE:211
NCS
AIX managed nodes, DCE:36
changing, AR:54–AR:56
description, AR:40
Net8, restricting access, AR:116
NetBios Sessions application, DCE:404
netcontool application, DCE:306
netop, CG:60
See also opc_adm; opc_op; operators
NetWare Agent Actions application, DCE:212
NetWare Config window, DCE:206
NetWare message group, AR:72
NetWare Performance window,
DCE:207–DCE:208
NetWare Tools
applications, DCE:209–DCE:212
window, DCE:208
NetWare. See Novell NetWare managed nodes
network
troubleshooting, HTTPS:190
Network Computing System. See NCS
Network Interfaces application, DCE:212
Network message group
MPE/iX, DCE:165
OVO, AR:72
Network Node Manager. See NNM
network security
DCE, AR:451–AR:456
overview, AR:450–AR:461
RPC authentication, AR:457–AR:458
SSH, AR:461
networking commands, excluding from
streamed jobs on MPE/iX managed nodes, DCE:161
nfile parameter, AR:38
nflocks parameter, AR:38
NFS troubleshooting, AR:443
NLM Files* application, DCE:213
NMA
2.1 agent, DCE:205
applications, DCE:212–DCE:214
description, DCE:204
monitoring performance, DCE:206
NMEV markers
generating new, DCE:169–DCE:170
mapping, DCE:166–DCE:169
$$NMEV\_APPL$$ variable, AR:164
$$NMEV\_CLASS$$ variable, AR:164
$$NMEV\_SEV$$ variable, AR:164
NNM
accessing from Java GUI
locally, AR:328–AR:329
remotely, AR:329–AR:330
collection stations with OVO agents, CG:487
on multiple management servers, CG:491
configuring access with command-line
tools, AR:332
DHCP synchronization, HTTPS:154
event correlation, CG:431
integrating applications into OVO,
AR:248–AR:253
limitations, AR:248
integrating into OVO, AR:247
SNMP event interceptor, CG:415
No Status Propagation display mode, CG:163,
CG:293
Node Advanced Options window, CG:244
node bank
add nodes, HTTPS:162
node certificates request, HTTPS:156
Node Communication Options window,
  CG:245
Node Config Report, AR:110
Node Group Bank window, AR:71
Node Group Report, AR:111
node groups
  adding, AR:71
  default, AR:71
  deleting, AR:71
  management server, AR:71
  modifying, AR:71
Node Groups Overview Report, AR:111
node hierarchies, CG:233–CG:234
node mapping tool, AR:334–AR:335
Node message attribute, AR:77
Node Reference Report, AR:111
Node Report, AR:111
Nodes folder
  colors, CG:71
  figure, CG:71
  groups, CG:71
  layout groups, CG:71
  overview, CG:71–CG:72
Nodes Overview Report, AR:111
nodes. See managed nodes; node groups: node hierarchies
  non-sequential conditions, CG:338
Normal message severity level, AR:74
nosec option, AR:322, AR:325
notification, CG:475
notification service
  concepts, AR:265
  configuring, AR:268
  parameters, AR:270
  writing scripts and programs,
    AR:266–AR:267
notification services
  forwarding messages, AR:133
notification system
  messages, CG:475–CG:476
notification, message event, CG:133
Novell NetWare managed nodes
  APIs, DCE:220–DCE:221
  applications
    NetWare Tools, DCE:209–DCE:212
    NMA, DCE:212–DCE:214
    overview, DCE:204–DCE:214
  assigning passwords, AR:470
default operator, DCE:218
directory structure, DCE:217
file locations, DCE:217
hardware requirements, DCE:187
include file, DCE:222
installation
  process, DCE:194–DCE:195
  requirements, DCE:187–DCE:189
  tips, DCE:190–DCE:193
installing agents, DCE:196–DCE:201
libraries, DCE:222–DCE:223
makefile, DCE:223
NMA
  2.1 agent, DCE:205
  applications, DCE:212–DCE:214
  description, DCE:204
  monitoring performance, DCE:206
organization, DCE:217–DCE:219
overview, DCE:185–DCE:223
preconfigured elements, DCE:202–DCE:214
removing agents, DCE:201
scripts and programs, DCE:215–DCE:216
SNMP event interceptor, DCE:203
software requirements, DCE:187–DCE:189
system resource files, DCE:218
windows
  NetWare Config, DCE:206
  NetWare Performance, DCE:207–DCE:208
  NetWare Tools, DCE:208
NS node name mapping, DCE:178–DCE:181
NT. See Windows NT/2000 managed nodes
NT_DWN_SMS_CLIENT_CONFIG_MANAGER monitor, DCE:445
NT_DWN_SMS_EXECUTIVE monitor, DCE:445
NT_DWN_SMS_HIERARCHY_MANAGER monitor, DCE:445
NT_DWN_SMS_INVENTORY_AGENT monitor, DCE:445
NT_DWN_SMS_PACKAGE_COMMAND_MANAGER monitor, DCE:445
NT_DWN_SMS_SITE_CONFIG_MANAGER monitor, DCE:445
NT_DWN_SMS_TRAP_FILTER monitor, DCE:445
NT_UP_SMS_CLIENT_CONFIG_MANAGER monitor, DCE:445
NT_UP_SMS_CLIENT_CONFIG_MONITOR monitor, DCE:445

Master Index

NT_UP_SMS_EXECUTIVE monitor, DCE:445
NT_UP_SMS_HIERARCHY_MANAGER monitor, DCE:445
NT_UP_SMS_INVENTORY_AGENT monitor, DCE:445
NT_UP_SMS_PACKAGE_COMMAND_MANAGER monitor, DCE:445
NT_UP_SMS_SITE_CONFIG_MANAGER monitor, DCE:445
NT_UP_SMS_TRAP_FILTER monitor, DCE:445

O

O message attribute, AR:76
<$O$> variable, AR:167
oareqhdl file, AR:354
Object message attribute, AR:77
object names, localizing, AR:313
object pane
   figures
      enabling, CG:201
      main window, CG:69
      popup menu, CG:112
folders
   Applications, CG:75
   Filter Settings, CG:76–CG:77
   Message Groups, CG:73–CG:74
   Nodes, CG:71–CG:72
   URL Shortcuts, CG:78
moving, CG:199
overview, CG:69–CG:70
popup menus, CG:112
saving message browser to, CG:214
showing, CG:201
object status, reviewing, CG:164
object tree, searching
   overview, CG:132
objects. See monitoring
ODI Info application, DCE:213
offline backups, AR:489
olh_About_Server_Config, DCE:412
olh_About_Server_Stats, DCE:413
olh_About_Shares, DCE:414
online documentation
   figure, CG:85
Online Help workspace, CG:85
OpC message group, AR:72
opc process, AR:349
OPC_ACCEPT_CTRL_SWITCH_ACKN parameter, AR:139
OPC_ACCEPT_CTRL_SWITCH_MSGS parameter, AR:139
OPC_ACCEPT_NOTIF_MSSGS parameter, AR:139
opc_adm, CG:56–CG:57
See also netop; opc_op; operators
OPC_AUTO_DEBUFFER parameter, AR:132
 opc_brc_history file, CG:176
$OPC_BRC_HISTSIZE variable, CG:176
$OPC_CUSTOM(name) variable, AR:174
$OPC_ENV(env variable) variable, AR:160, AR:171
$OPC_EXACT_SELECTED_NODE_LABEL variable, AR:174
$OPC_EXT_NODES variable, AR:171
OPC_FORW_CTRL_SWITCH_TO_TT parameter, AR:139
OPC_FORW_NOTIF_TO_TT parameter, AR:139
opc_get_ems_resource monitor executable, DCE:113
<$OPC_GUI_CLIENT> variable, AR:160
$OPC_GUI_CLIENT variable, AR:174
$OPC_GUI_CLIENT_WEB variable, AR:174
<$OPC_MGMTSV> variable, AR:158, AR:161
$OPC_MGMTSV variable, AR:171
$OPC_MSG.ACTIONS.AUTOMATIC variable, AR:175
$OPC_MSG.ACTIONS.AUTOMATIC.ACKNOWLEDGE variable, AR:175
$OPC_MSG.ACTIONS.AUTOMATIC.ANNOTATION variable, AR:176
$OPC_MSG.ACTIONS.AUTOMATIC.ANNOUNCE variable, AR:176
$OPC_MSG.ACTIONS.AUTOMATIC.COMMAND variable, AR:176
$OPC_MSG.ACTIONS.AUTOMATIC.COMMAND[n] variable, AR:176
$OPC_MSG.ACTIONS.AUTOMATIC.STATUS variable, AR:176
$OPC_MSG.ACTIONS.OPERATOR variable, AR:176
$OPC_MSG.ACTIONS.OPERATOR.ACKNOWLEDGE variable, AR:177
$OPC_MSG.ACTIONS.OPERATOR.ANNOUNCE variable, AR:177
$OPC_MSG.ACTIONS.OPERATOR.COMMAND variable, AR:177
$OPC_MSG.ACTIONS.OPERATOR.COMMAND[n] variable, AR:177
Master Index

syntax, AR:512
opcmacll command, AR:543
opcmapnode command, AR:332
opcmon command, CG:397
opcmon(1) command, AR:543
opcmon(3) API, AR:543
opcmona process, AR:356
opcmsg
templates
    HP-UX (OVO), DCE:97
    Solaris (OVO), DCE:287
opcmsg for OV Performance message
template, AR:220
opcmsg(1) command
description, AR:543
flow, CG:391
opcmsg(3) API
description, AR:543
EMS, DCE:113
flow, CG:391
opcmsga process, AR:357
opcmsgi process, AR:357
opcmsgm process, AR:350
opcmsgmgr process, AR:351
opcmsgma process, AR:351
opcnod
DHCP variables, HTTPS:153
opcswinfo, HTTPS:126
opctmpldwn, AR:470
opctrapi process, AR:357
opctts process, AR:351
opcttnsm process, AR:351
opcuiadm process, AR:352
opcuiop process, AR:352
opcuiopadm process, AR:352
opcuiwww process, AR:352
opcwam command, AR:493
Open Files application, DCE:213
opening
    Download Configuration Data window,
    AR:487
OpenView applications in Java GUI, AR:330–AR:332
integrating
    Ethernet Traffic HP as OV application,
    AR:250
    IP Activity Monitoring - Tables as OV
    service, AR:251
internal traps, DCE:99
maintaining, AR:504
OpenView applications, accessing, CG:156
OpenView Operations for Windows
configuring
    agent policy, AR:239
    agents for OVO management server,
    AR:232–AR:234
    OVO agents for management server,
    AR:228–AR:231
    servers to forward messages to OVO,
    AR:235–AR:240
exporting policies to OVO, AR:242
forwarding messages on management
server, AR:236
importing OVO templates, AR:241
interoperability with OVO, AR:227–AR:242
sending messages to management server,
AR:228
OpenView Operations. See OVO
OpenView Performance Agent. See OVPA
Open. Active Details Report, AR:111
Open. Active Message Report, AR:111
operating systems
    AIX, DCE:31–DCE:65
    HP-UX
        OVO, DCE:67–DCE:122
        OVPA, AR:205–AR:224
    Linux, DCE:125–DCE:152
    MPE/iX, DCE:153–DCE:183
    Novell NetWare, DCE:185–DCE:223
    Sequent DYNIX, DCE:225–DCE:238
    SGI IRIX, DCE:239–DCE:252
    SINIX RM/Reliant, DCE:253–DCE:270
    Solaris
        OVO, DCE:271–DCE:313
        OVPA, AR:205–AR:224
        patches, DCE:279
    Tru64 UNIX, DCE:315–DCE:353
Operator History Messages Report, AR:111
operator instructions
    reading, CG:168–CG:169
Operator Overview Report, AR:111
Operator Pending Messages Report, AR:111
Operator Report, AR:111
operator-initiated actions
    annotations, CG:167
corrective actions, CG:393
process, CG:53–CG:54
protecting, AR:471
reviewing, CG:167
starting, CG:167
verifying, CG:167
operators
See also netop; opc_adm; opc_op; template
administrators; users; OVO
administrator
accessing GUI
Java, AR:465
Motif, AR:464
assigning applications, AR:245
changing
names, AR:462
passwords, AR:462
default
AIX, DCE:61
HP-UX, DCE:108
Linux, DCE:148–DCE:149
MPE/iX, DCE:177
Novell NetWare, DCE:218
Sequent DYNIX, DCE:235
SGI IRIX, DCE:249
SINIX RM/Reliant, DCE:267
Solaris, DCE:296
Tru64 UNIX, DCE:348
Windows NT/2000, DCE:430
defaults
system, CG:188
description, CG:59–CG:61
enabling
to control OVO agents, AR:252–AR:253
to manage IP networks in IP map, AR:249
mathematical, CG:338–CG:339
multiple, CG:55
reports
customized, AR:115
preconfigured, AR:114
saving output, AR:463
security, AR:462–AR:474
types, CG:60
windows, CG:60–CG:61
optimizing
message filtering, CG:355–CG:378
Optional ownership mode, CG:162, CG:294
<$OPTION(N)> variable, AR:158
options
Automatic (De-)Installation, AR:51
organizing
overview, CG:337–CG:338
sequence, CG:355
managed nodes
AIX, DCE:60–DCE:61
HP-UX, DCE:106–DCE:109
Linux, DCE:147–DCE:149
MPE/iX, DCE:177–DCE:181
Novell NetWare, DCE:217–DCE:219
overview, CG:227–CG:250
Sequent DYNIX, DCE:234–DCE:236
SGI IRIX, DCE:248–DCE:250
SINIX RM/Reliant, DCE:266–DCE:268
Solaris, DCE:295–DCE:297
Tru64 UNIX, DCE:347–DCE:349
message groups
overview, CG:251–CG:252
template groups, CG:310–CG:311
organizing Message Groups folder, CG:74
original message text, reviewing, CG:146
OS message group
MPE/iX, DCE:165
OVO, AR:72
outage template, AR:119
outages, scheduling, CG:441
output
EMS Resources application, DCE:118
operator, CG:222, AR:463
OVO administrator, AR:464
Output message group
MPE/iX, DCE:165
OVO, AR:72
OV Performance Agent template group,
AR:220
OV Performance Manager template group,
AR:220
ovbackup.ovp command, AR:494–AR:495
ovc, HTTPS:37
ovcert, HTTPS:39
ovconfget, HTTPS:37
OvCoreID, HTTPS:156
ovcoreid, HTTPS:37
OVDataDir, HTTPS:36
OVInstallDir, HTTPS:36
OVKey licenses
  advantages, AR:510
  replacing Instant On, AR:510
OVnlm_exit() API, DCE:220
OVnlm_init() API, DCE:220
OVO
  applications, CG:235
  character code conversion, AR:298–AR:304
  communication, AR:347–AR:348
  concepts
    client-server, CG:33–CG:35
    user, CG:55–CG:61
  configuring
    notification services, AR:263–AR:270
    to accept messages forwarded from
      OpenView Operations for Windows,
      AR:237–AR:239
    trouble ticket system, AR:263–AR:270
  database tables and tablespaces, AR:547
  defaults
    administrator, CG:191
    description, CG:33–CG:38
    Distributed Event Interception, DCE:99
      configuring, DCE:100
      description, DCE:99
    event interceptor, CG:431
  exporting templates to OpenView
    Operations for Windows, AR:241
  features, CG:17
  filtering internal error messages, CG:426, AR:384
  functionality, CG:39–CG:43
  importing OpenView Operations for
    Windows policies, AR:242
  improving performance, AR:372–AR:373
  installing configuration on managed nodes,
    AR:187–AR:203
  integrating applications
    actions, AR:255–AR:256
    Application Desktop, AR:246–AR:247
    broadcast commands, AR:254
    components, AR:245
  HP applications, AR:245
  HP OpenView plug-in, AR:246
    monitoring applications, AR:257
    overview, AR:243–AR:262
  OVO applications, AR:246
  integrating SMS, DCE:443–DCE:444
  interoperability
    OpenView Operations for Windows,
    AR:227–AR:242
    overview, AR:225–AR:242
    language support, AR:273–AR:313
    maintaining, CG:219–CG:301,
      AR:483–AR:540
    man pages, AR:558
    mapping file problems, DCE:180
    MC/ServiceGuard support, DCE:121
    message interface, CG:391–CG:392
    monitoring, CG:131
    other languages, AR:312
    overview, CG:31–CG:61
  process
    groups, AR:459
    names, AR:459
    processes, AR:345–AR:365
  security
    auditing, AR:475–AR:478
    levels, AR:460
    methods, CG:226
    operations, AR:462–AR:474
    overview, AR:445–AR:481
    OVO processes, AR:459–AR:460
  Spanish language, AR:307
  starting from operator GUI, CG:222
  Sun Enterprise Cluster support, DCE:312
  Sun Management Center integration,
    DCE:311
  tasks, CG:44–CG:54
  troubleshooting, AR:375–AR:384
    server, AR:388–AR:389
  tuning performance, AR:370–AR:374
  updating configuration on managed nodes,
    AR:187–AR:203
  variables, CG:174
  versions, AR:376–AR:377
  OVO Add Node window, CG:242–CG:245
  OVO administrator
See also administrative rights; operators; 
template administrators; users 
changing responsibility matrix, CG:224 
description, CG:56—CG:57 
environment, CG:221—CG:224 
GUI 
access, AR:464 
description, CG:222 
message browser, CG:223—CG:224 
reports 
customized, AR:113 
preconfigured, AR:110 
responsibility matrix, CG:224 
saving, AR:464 
OVO agents 
See also OVO 
activating on Solaris managed nodes 
command line, DCE:282 
GUI, DCE:283 
configuration files 
location, AR:362 
types, AR:361 
configuring OpenView Operations for 
Windows management server, 
AR:228—AR:231 
de-installing from managed nodes 
AIX, DCE:54 
automatically, AR:62—AR:63 
HP-UX, DCE:96 
Linux, DCE:140—DCE:141 
manually, AR:63 
MPE/iX, DCE:163 
Sequent DYNIX, DCE:230 
SGI IRIX, DCE:244 
SINIX RM/Reliant, DCE:262 
Solaris, DCE:285 
Tru64 UNIX, DCE:334 
Windows NT/2000, DCE:385 
distributing configuration to managed 
nodes, AR:189 
ensuring operators to control, 
AR:252—AR:253 
HACMP, DCE:46 
installation 
managed nodes, AR:35—AR:56 
reasons not to install, CG:237 
requirements, AR:37—AR:40 
script, AR:48 
tips, AR:41—AR:47 
installing on managed nodes 
AIX, DCE:41—DCE:53 
HP-UX, DCE:85—DCE:92 
Linux, DCE:136—DCE:139 
MPE/iX, DCE:163 
Novell NetWare, DCE:196—DCE:201 
Sequent DYNIX, DCE:230 
SGI IRIX, DCE:244 
SINIX RM/Reliant, DCE:261 
Solaris, DCE:280—DCE:281 
Sun Enterprise E10000, 
DCE:309—DCE:310 
Tru64 UNIX, DCE:328 
Windows NT/2000, DCE:361—DCE:384 
MANAGING, AR:64—AR:66 
monitoring 
IP devices, CG:486 
objects, CG:395—CG:400 
reconfiguring on regional management 
servers, CG:461 
removing from managed nodes 
AIX, DCE:54 
Linux, DCE:141 
Novell NetWare, DCE:201 
SGI IRIX, DCE:244 
SINIX RM/Reliant, DCE:262 
Solaris, DCE:286 
SSH installation method, AR:57—AR:61 
synchronizing commands with character 
set, AR:286 
updating on managed nodes, AR:48—AR:56 
versions 
description, AR:64 
displaying available, AR:65 
displaying installed, AR:65 
removing, AR:66 
with NNM collection stations, CG:487 
on multiple management servers, CG:491 
OVO Application Bank window 
EMS resource hierarchy, DCE:118—DCE:119 
OVO Error Report, AR:112, AR:114 
OVO in a Cluster environment 
architecture, DCE:451 
preconfigured elements, DCE:463
Master Index

troubleshooting, DCE:459–DCE:462
OVO management server
certificate troubleshooting, HTTPS:208
communication troubleshooting, HTTPS:204
OvCoreIds, HTTPS:209
OVO Message Group Bank window, CG:251
OVO Node Bank window, CG:229–CG:230
OVO Node Hierarchy Bank window,
CG:231–CG:235
OVO Node Hierarchy window, CG:228
ovoareqsdr process, AR:349
OVOPC-CLT agent filesets
  English only, DCE:82
generic, DCE:82
OVPA
  AIX, AR:207
  applications, AR:218
  customizing, AR:209
data
  analyzing, AR:208
  integrating, AR:208
  logging, AR:208
de-installing from managed nodes, AR:216
description, AR:208–AR:209
documentation
downloading, AR:223
PDFs, AR:223
viewing, AR:223
hardware requirements, AR:210
HP-UX, AR:205–AR:224
installation requirements, AR:210–AR:211
installing on managed nodes,
AR:212–AR:215
overview, AR:205–AR:224
software requirements, AR:210–AR:211
Solaris, AR:205–AR:224
templates, AR:220–AR:222
Tru64 UNIX, AR:207
ovpolicy, HTTPS:38
ovrc, HTTPS:38
ovrestore.ovpl command, AR:495–AR:497
ownership
display modes, CG:163, CG:292–CG:293
modes, CG:162, CG:293–CG:295
Ownership policy, CG:135
owning messages, CG:292

P
PAM, authentication, AR:466
panes and areas
  moving, CG:199
  showing and hiding, CG:201–CG:203
parallel configuration servers, HTTPS:32
parameters
  See also variables
  kernel, AR:38
message buffering, AR:132
notification service, AR:270
opclic command, AR:512–AR:513
scheduled outages
  syntax, AR:131
SYSSTART.PUB.SYS, DCE:161
templates
  message forwarding, AR:139
  scheduled outages, AR:131
service hours, AR:131
time zone string, AR:136
trouble ticket system, AR:270
passwd option, AR:322, AR:325
passwords
  assigning, AR:468–AR:470
  changing, CG:186, AR:462
  controlling, AR:462
DCE nodes, AR:467–AR:468
  root, AR:48
patches, Solaris, DCE:279
pattern matching
  condition examples, CG:339–CG:340
  mathematical operators, CG:338–CG:339
  messages, CG:338–CG:346
  returning node names, AR:334
  syntax, CG:341–CG:343
  without case-sensitivity, CG:339
pattern-matching options
  setting defaults, CG:325
PDF documentation
OVPA, AR:223
pending messages browser
  See also active message browser; filtered
  message browser; history message
  browser; message browser
investigating problems, CG:159
overview, CG:99
unbuffering messages, CG:159

280

AR  OVO Administrator's Reference
DCE  OVO DCE Agent Concepts and Configuration Guide
CG  OVO Concepts Guide
HTTPS  OVO HTTPS Agent Concepts and Configuration Guide
Master Index

perfbd monitor template, AR:221
PerfMon Objs application, DCE:405
performance
agent, HTTPS:33
improving
database, AR:371
Motif GUI startup, AR:374
OVO, AR:372–AR:373
SNMP management platform,
AR:370–AR:371
Java GUI, AR:342–AR:343
monitoring, CG:37
NMA, DCE:206
optimizing, CG:355–CG:356
troubleshooting, HTTPS:35
tuning, AR:370–AR:374
Performance Agent. See OVPA
Performance message group
MPE/iX, DCE:165
OVO, AR:73
performance metrics
about, CG:398
configuring, CG:399
monitoring, CG:398
Perl interpreter
AIX, DCE:65
HP-UX, DCE:122
Linux, DCE:152
Solaris, DCE:313
Tru64 UNIX, DCE:353
Windows NT/2000, DCE:448
permissions
file access, AR:463
GUI, AR:464–AR:465
setting
group, AR:463
setting file, AR:463
Personal Filters, CG:77
physical node, HTTPS:146
Physical Terminal application, DCE:173
pids file, AR:354, AR:359
ping
application, HTTPS:184
pipe files
managed nodes, AR:358–AR:359
management server, AR:353–AR:354
platform, HTTPS:157
plug-in, HP OpenView application, AR:246
point-to-point problems, AR:436
policies
assigning to virtual nodes, HTTPS:150
changing WM1 default name, AR:240
de-assigning from virtual nodes, HTTPS:150
deploying policies to virtual nodes,
HTTPS:151
importing OpenView Operations for
Windows policies into OVO, AR:242
manual installation, HTTPS:91
message escalation, CG:453
messages, CG:134
modifying policies on virtual nodes,
HTTPS:151
policy management, HTTPS:90
polling intervals
MIB objects, CG:396
programs, CG:396
popup menus
browser pane, CG:115
customizing, CG:206–CG:207
object pane, CG:112
overview, CG:110
shortcut bar, CG:111
workspace pane, CG:113
port option, AR:325
position controls
figures
enabling, CG:198
main window, CG:109
hiding, CG:198
overview, CG:109
showing, CG:198
PRC authentication, AR:454
preconfigured
elements, AR:71–AR:108
AIX, DCE:55–DCE:57
HP-UX (OVO), DCE:97–DCE:102
HP-UX (OVPA), AR:218–AR:222
Linux, DCE:142–DCE:143
MPE/iX, DCE:164–DCE:174
Novell NetWare, DCE:202–DCE:214
Sequent DYNIX, DCE:231
SGI IRIX, DCE:245
SINIX RM/Reliant, DCE:263
Solaris (OVO), DCE:287–DCE:292
Solaris (OVPA), AR:218–AR:222

281
profiles
management, CG:457
user, CG:56
$PROG$ variable, AR:169
Program Neighbourhood service, DCE:436
programs
accessing
HP-UX, AR:465
MPE/iX, AR:465
distribution
AIX, DCE:58–DCE:59
HP-UX, DCE:103–DCE:105
Linux, DCE:144–DCE:146
MPE/iX, DCE:175–DCE:176
Novell NetWare, DCE:215–DCE:216
overview, AR:190–AR:194
requirements, AR:190
Sequent DYNIX, DCE:232–DCE:233
SGI IRIX, DCE:246–DCE:247
SINIX RM/Reliant, DCE:264–DCE:265
Solaris, DCE:293–DCE:294
tips, AR:190–AR:193
Tru64 UNIX, DCE:337–DCE:338
monitors, CG:395
notification service, AR:266–AR:267
security, AR:465
trouble ticket system, AR:266–AR:267
prompt_for_activate option, AR:325
properties, changing default types of all messages forwarded to OVO, AR:240
protecting
automatic actions, AR:471
configuration distribution, AR:470
operator-initiated actions, AR:471
remote actions, AR:471–AR:474
shell scripts, AR:471
template distribution, AR:470
proxies, HTTPS:138
configuring, HTTPS:140
dual-homed host, HTTPS:139
manual agent software installation, HTTPS:143
multi-homed host, HTTPS:139
on management server, HTTPS:144
single-homed host, HTTPS:139
syntax, HTTPS:142
pvalarmd monitor template, AR:222
Q
queue files
managed nodes, AR:358–AR:359
management server, AR:353–AR:354
removing, AR:500
security, AR:474
Queues application, DCE:213
quick filters, accessing, CG:214
R
$<R>$ variable, AR:167
$<r>$ variable, AR:167
ratio, management hierarchy setup, CG:458
Reactivate alarmdef application, AR:218
reading operator instructions,
CG:168–CG:169
Reboot application, DCE:408
reconfiguring
management server after changing hostname or IP address,
OVO agents on regional management servers, CG:461
SSP
snmpd daemon, DCE:307
templates, DCE:309, DCE:310
reconnect_interval option, AR:326
reconnect_timeout option, AR:326
recovering
See also recovery tools
configuration data after automatic backup,
AR:498–AR:500
database to latest state, AR:498–AR:499
recovery tools, AR:488
See also recovering
redistributing scripts to all managed nodes,
AR:488
redo logs, creating another set, AR:503
reducing number of messages,
CG:357–CG:378
refresh interval
changing, CG:193
refresh_interval option, AR:322, AR:326
Reg Viewer application, DCE:409
regional management servers
configuring, CG:461–CG:462
Master Index

description, CG:458
managed nodes, CG:461–CG:462
reconfiguring OVO agents, CG:461
registering problems, CG:39
regroup conditions
See also regrouping messages
defining, CG:382
eamples, CG:383
Regroup Conditions window, CG:382
regrouping messages
See also regroup conditions
description, CG:312
overview, CG:381–CG:383
Reliant. See SINIX RM/Reliant managed nodes
remote access
See also remote actions
applications, AR:467
broadcast commands, AR:467
I/O applications, AR:467
remote actions
See also remote access
eample, AR:472
protecting, AR:471–AR:474
security mechanisms, AR:473–AR:474
remote control, HTTPS:98
remote host equivalence, establishing, DCE:308
remote installation
Linux, DCE:136
removing
See also de-installing; installing
DCE
AIX, DCE:41
SINIX RM/Reliant, DCE:261
Tru64 UNIX, DCE:327
OVO agents, AR:66
AIX, DCE:54
Linux, DCE:141
Novell NetWare, DCE:201
SGI IRIX, DCE:244
SINIX RM/Reliant, DCE:262
Solaris, DCE:286
queue files, AR:500
Removing Older Agents, DCE:141
rep_server monitor template, AR:221
replacing Instant On licenses with OVKey licenses, AR:510
reporting errors
GUI Error Dialog Box, AR:382–AR:383
message browser, AR:381
overview, AR:380–AR:384
stderr and stdout devices, AR:383
reports
administrator
customized, AR:113
preconfigured, AR:110
configuring timeouts, AR:109
database, AR:109–AR:116
defining printer, AR:109
generating, CG:40
Internet, AR:109
operator
customized, AR:115
preconfigured, AR:114
security, AR:116
statistical, AR:115
trend analysis, AR:115
requirements. See distribution; installation requirements
rerunning automatic actions, CG:165
reset message, sending automatically, CG:367–CG:369
resetting
events
HACMP 4.2.2, DCE:51
HACMP 4.3.1, DCE:51–DCE:52
IP alias for HACMP agents in GUI, DCE:50
resolving message attributes, CG:323
resource instances, viewing in EMS GUI, DCE:116
resource requirements, HTTPS:32
RESPMGRCONFIG keyword, AR:119
responding to messages, CG:50
responsibility
See also responsible managers
distributing in competence centers,
CG:450–CG:451
domain hierarchy management,
CG:458–CG:459
management server
delegating, CG:468
switching, CG:467–CG:469
operator matrix, CG:224
responsible managers
See also responsibility
configuration file
  creating, CG:463
  distributing, CG:464
  configuring, CG:463–CG:471
templates
  managed nodes, CG:464
  syntax, AR:125
Restart PA Servers application, AR:218
Restart Perf Agt application, AR:218
restore
certificate, HTTPS:212
restoring database, AR:498
restricting
  See also restrictions
database access, AR:116
Net8 access, AR:116
web reporting, AR:116
restrictions
  See also restricting
OVO access, CG:56
results, action, CG:164
reversing manager switch, CG:468
reviewing
  acknowledgements, CG:184
  annotations
    actions, CG:164
    messages, CG:181
  automatic actions, CG:165
  messages
    attributes, CG:144
    details, CG:95
    groups, CG:252
  object status, CG:164
  operator-initiated actions
    annotations, CG:167
    overview, CG:167
RM/Reliant. See SINIX RM/Reliant managed nodes
roles, user, CG:55
ROMAN8, converting managed node files, AR:300
root
  passwords, AR:48
  user, AR:466
root certificate, HTTPS:51
deployment, HTTPS:54
update, HTTPS:54
RPC
authentication, AR:457–AR:458
configuring in OVO, AR:458
OVO example, AR:458
login context, AR:457
server ticket
  description, AR:457
  verifying, AR:457
time out, HTTPS:188
troubleshooting, AR:427
rqsdbf file, AR:354
rqsp pipe file, AR:354
rqsq queue file, AR:354
rules, message target, CG:465–CG:466
Running Software® application, DCE:213
runtime problems
  all managed nodes, AR:401–AR:415
  managed node directories, AR:507
  MPE/iX managed nodes, AR:420–AR:426
  UNIX managed nodes, AR:416–AR:419
S
S message attribute, AR:75
<$S$> variable, AR:167
<$s$> variable, AR:168
SAM
ASCII, DCE:101
Motif, DCE:101
  OVO Application Bank window, DCE:118–DCE:119
sam command, DCE:101
Save Browser Filter Settings dialog box
  figure, CG:213
saving
  console settings
    figure, CG:195
    overview, CG:195–CG:197
customized message browser layout, CG:218
message browser filter
  object pane, CG:214
  settings, CG:212–CG:213
output
  operator, CG:222, AR:463
  OVO administrator, AR:464
scalability
  multiple management servers, CG:443–CG:491
  scenarios, CG:484–CG:491
scanning messages, CG:134
scenarios
automating standard, CG:364
scalability
multiple management servers, CG:489–CG:490
multiple management servers with OVO agents and NNM collection stations, CG:491
NNM collection station with OVO agents, CG:487–CG:488
OVO agents monitoring IP devices, CG:486
single management server, CG:484–CG:485
scheduled outages
configuring, CG:442
defining, CG:441
overview, CG:441
template
examples, AR:153
location, AR:130
parameters, AR:131
syntax, AR:128–AR:130
scheduling templates, AR:130–AR:136
scopeux monitor template, AR:221
scripts
customized, AR:191
distributing, AR:190–AR:194
distribution
AIX, DCE:58–DCE:59
HP-UX, DCE:103–DCE:105
Linux, DCE:144–DCE:146
MPE/iX, DCE:175–DCE:176
Novell NetWare, DCE:215–DCE:216
requirements, AR:190
Sequent DYNIX, DCE:232–DCE:233
SGI IRIX, DCE:246–DCE:247
SINIX RM/Reliant, DCE:264–DCE:265
Solaris, DCE:293–DCE:294
tips, AR:190–AR:193
Tru64 UNIX, DCE:337–DCE:338
ito_restore.sh, AR:497
notification service, AR:266–AR:267
redistributing, AR:488
shell, protecting, AR:471
trouble ticket system, AR:266–AR:267
versions, AR:190
SD-UX
See also HP-UX managed nodes
creating software depot on remote node, DCE:87–DCE:88
enabling, DCE:89
installing OVO agents from depot node, DCE:86
from SD-UX depot, DCE:89
manually from depot, DCE:92
manually from tape file, DCE:91
overview, DCE:86–DCE:89
searching object tree
overview, CG:132
second disk, moving database control files, AR:502
secondary manager
enabling actions, CG:468
specifying, CG:460
switching responsibility, CG:467–CG:468
SECONDARYMANAGERS keyword, AR:120
secure_port option, AR:326
securing environment, CG:225–CG:226
security
alternative users, HTTPS:70
agent profile, HTTPS:77
changing default port, HTTPS:76
comparison with DCE agents, HTTPS:84
configuring the management server, HTTPS:75
installation, HTTPS:73
limitations, HTTPS:71
patching, HTTPS:80
preparation, HTTPS:72
sudo, HTTPS:81
upgrading, HTTPS:80
auditing, AR:475–AR:478
certificate client, HTTPS:48, HTTPS:53
certificate server, HTTPS:48, HTTPS:52
merging, HTTPS:56
multiple, HTTPS:55, HTTPS:59
sharing, HTTPS:61
certificates, HTTPS:51
certification authority, HTTPS:52
components, HTTPS:48
database, AR:465
exception warnings, AR:343
key store, HTTPS:48
levels, DCE:166
managed nodes, CG:247
network
  DCE, AR:451–AR:456
  overview, AR:450–AR:461
  RPC authentication, AR:457–AR:458
operations
  accessing OVO, AR:462
  overview, AR:462–AR:474
overview, AR:445–AR:481
OVO, CG:226
  levels, AR:460
  process, AR:459–AR:460
  processes, AR:363–AR:365
  program, AR:465
remote actions, AR:473–AR:474
reports, AR:116
root certificate, HTTPS:51
  deployment, HTTPS:54
  update, HTTPS:54
SSH, AR:461
types, AR:447
Security message group
  MPE/iX, DCE:165
  OVO, AR:73
Sel. Active Details Report, AR:114
Sel. Active Messages Report, AR:114
Sel. History Details Report, AR:114
Sel. History Messages Report, AR:114
Sel. Pending Details Report, AR:114
Sel. Pending Messages Report, AR:114
selecting
  conditions, CG:338
  message generation policy, CG:402–CG:405
  threshold types, CG:401
semmsn parameter, AR:38
Send Message application, DCE:440
sending
  messages to management server
    OpenView Operations for Windows, AR:228
    OVO, AR:232
  reset message automatically,
    CG:367–CG:369
Sequent DYNIX managed nodes
default operator, DCE:235
dep-installing agents, DCE:230
directory structure, DCE:234
file locations, DCE:234
hardware requirements, DCE:227
install file, DCE:237
installation
  requirements, DCE:227–DCE:228
  tips, DCE:229
installing agents, DCE:230
libraries, DCE:237–DCE:238
makefile, DCE:238
organization, DCE:234–DCE:236
overview, DCE:225–DCE:238
preconfigured elements, DCE:231
scripts and programs, DCE:232–DCE:233
SNMP event interceptor (not supported), DCE:231
software requirements, DCE:227–DCE:228
system resource files, DCE:236
sequential conditions
  description, CG:355
  selecting, CG:338
Server Config application, DCE:412
server option, AR:322
Server Stats application, DCE:413
server ticket, RPC, AR:457
Servers application, DCE:440
servers. See management server; managers
Service Desk, AR:265
service hours, CG:99
configuring, CG:442
defining, CG:440
overview, CG:439–CG:440
template
  examples, AR:152
  location, AR:130
  parameters, AR:131
  syntax, AR:128, AR:130
Service Navigator
  finding impacted services, CG:156
Service Navigator man pages, AR:563
service template, AR:119
services
  ICA Browser, DCE:435
  OV Service, AR:251
  Program Neighbourhood, DCE:436
Master Index

Services workspace
  finding impacted Service Navigator services, CG:156
  overview, CG:82
Sessions application, DCE:441
Set Parameters® application, DCE:213
setting
  character set
    GUI, AR:277–AR:283
    managed nodes, AR:287
    management server, AR:276
  community name
    opinfo file, AR:433
    SNMP daemon configuration file, AR:434
  file permissions, AR:463
  group permissions, AR:463
IP aliases for HACMP agents
  AIX 4.3, DCE:48
language
  managed nodes, AR:286
  management server, AR:275
setting up
  customized job stream facility on MPE/iX
    managed nodes, DCE:162
management
  hierarchies, CG:458
  server defaults, CG:446
message
  conditions, CG:333–CG:337
  defaults, CG:324–CG:325
  node hierarchy, CG:233
  threshold monitoring, CG:409–CG:410
  time intervals in time templates, CG:466
settings
  compression, CG:373
  node defaults, CG:246
  settings, console, CG:195–CG:197
severity
  message coloring, CG:139–CG:141
  viewing in Message Dashboard,
    CG:151–CG:155
severity messages
  evaluating, CG:318
  levels, AR:74–AR:75
Severity policy, CG:134
severity_label option, AR:326
SGI IRIX managed nodes
default operator, DCE:249
de-installing agents, DCE:244
directory structure, DCE:248
file locations, DCE:248
hardware requirements, DCE:241
include file, DCE:251
installation
  requirements, DCE:241–DCE:242
  tips, DCE:243
installing agents, DCE:244
libraries, DCE:251–DCE:252
logfile templates, DCE:245
makefile, DCE:252
organization, DCE:248–DCE:250
overview, DCE:239–DCE:252
preconfigured elements, DCE:245
removing agents, DCE:244
scripts and programs, DCE:246–DCE:247
SNMP event interceptor (not supported),
  DCE:245
software requirements, DCE:242
system resource files, DCE:250
Shares application, DCE:414
sharing a certificate server, HTTPS:61
sharing message control, CG:473
shell script syntax, AR:267
shell scripts, protecting, AR:471
Shift JIS
  converting managed nodes to, AR:306
  processing management server files, AR:302
shmmax parameter, AR:38
shortcut bar
  customizing, CG:204
  figures
    disabling, CG:202
    enabling, CG:201
    main window, CG:67
    popup menu, CG:111
  hiding, CG:201
  moving, CG:199
  overview, CG:67–CG:68
  popup menus, CG:111
  showing, CG:201
shortcut_tree_icon_width option, AR:326
shortcuts, assigned by the OVO administrator, CG:191
Show Drivers application, DCE:415
Show Services application, DCE:416
Show Users application, DCE:419
show_at_severity option, AR:326
showing
message browser columns, CG:217
panes and areas, CG:201–CG:203
position controls, CG:198
Siemens-Nixdorf. See hardware; SINIX
RM/Reliant managed nodes
Silicon Graphics Indigo. See hardware; SGI
IRIX managed nodes
single-homed host, HTTPS:139
SINIX RM/Reliant managed nodes
DCE
configuring, DCE:260
removing, DCE:261
OVO
default operator, DCE:267
de-installing agents, DCE:262
directory structure, DCE:266
file locations, DCE:266
hardware requirements, DCE:255
installation requirements,
DCE:255–DCE:256
installation tips, DCE:257–DCE:259
installing agents, DCE:261
libraries, DCE:269–DCE:270
makefile, DCE:270
organization, DCE:266–DCE:268
overview, DCE:253–DCE:270
preconfigured elements, DCE:263
removing agents, DCE:262
scripts and programs, DCE:264–DCE:265
SNMP event interceptor (not supported),
DCE:263
software requirements, DCE:255–DCE:256
system resource files, DCE:268
size, message distribution list,
CG:477–CG:479
smit command, DCE:57
SMT User Interface, starting, DCE:57
SMS
integrating into OVO, DCE:443–DCE:444
integration, DCE:442–DCE:447
monitors, DCE:445
versions supported, DCE:442
SNMP
configuration file, AR:434

event interceptor
AIX, DCE:56
HP-UX (OVO), DCE:99–DCE:101
Linux (not supported), DCE:143
MPE/iX (not supported), DCE:171
Novell NetWare, DCE:203
Sequent DYNIX (not supported), DCE:231
SGI IRIX (not supported), DCE:245
SINIX RM/Reliant (not supported),
DCE:263
Solaris (OVO), DCE:289–DCE:291
Tru64 UNIX (not supported), DCE:335
events, CG:414–CG:421
traps
adding templates, CG:418
condition example, CG:420
defining template conditions,
CG:418–CG:419
forwarding, CG:416–CG:417
OpenView, DCE:99
overview, CG:414–CG:421
Sun Enterprise E10000, DCE:303
variables, AR:165–AR:168
well-defined, DCE:99
SNMP message group, AR:73
software
communication, AR:39–AR:40
debugging (de-)installation, AR:67–AR:68
installation, HTTPS:101
from clone images, HTTPS:128
manual, HTTPS:119
manual behind proxy, HTTPS:143
manually from package files, HTTPS:120
software requirements
OVO
AIX, DCE:33–DCE:36
HP-UX, DCE:70–DCE:75
Linux, DCE:128–DCE:132
MPE/iX, DCE:155–DCE:156
Novell NetWare, DCE:187–DCE:189
Sequent DYNIX, DCE:227–DCE:228
SGI IRIX, DCE:242
SINIX RM/Reliant, DCE:255–DCE:256
Solaris, DCE:274
Tru64 UNIX, DCE:318–DCE:320
Windows NT/2000, DCE:359–DCE:360
software sub-tree on management server
customer-specific, DCE:81
vendor-specific, DCE:80
Solaris managed nodes
See also Sun Clusters; Sun Enterprise
E10000; Sun Management Center; Sun
SPARCclassic; Sun SPARCserver; Sun
SPARCstation; Sun Ultra
OVO
activating agents, DCE:282–DCE:283
default operator, DCE:296
de-installing agents, DCE:285
directory structure, DCE:295
file locations, DCE:295
hardware requirements, DCE:273
include file, DCE:299
installation requirements,
DCE:273–DCE:276
installation tips, DCE:277–DCE:278
installing agents, DCE:280–DCE:281
libraries, DCE:298–DCE:300
logfile locations, AR:509
logfile templates, DCE:288
makefile, DCE:300
MC/ServiceGuard support, DCE:121
message templates, DCE:287
missing OS patches, DCE:279
organization, DCE:295–DCE:297
overview, DCE:271–DCE:313
preconfigured elements,
DCE:287–DCE:292
removing agents, DCE:286
scripts and programs, DCE:293–DCE:294
SNMP event intercepter,
DCE:289–DCE:291
software requirements, DCE:274
Sun Enterprise Cluster support, DCE:312
Sun Enterprise E10000,
DCE:301–DCE:310
Sun Management Center integration,
DCE:311
system resource files, DCE:296
template groups, DCE:287
OVPA
de-installing, AR:216
installation requirements, AR:210–AR:211
installing, AR:212–AR:215
overview, AR:205–AR:224
preconfigured elements, AR:218–AR:222
template groups, AR:220–AR:222
Solaris node group, AR:71
Solaris template group, DCE:287
solutions, documenting, CG:40, CG:178
solving problems, CG:39
accessing terminal, CG:177
adding OVO variables, CG:174
applications, CG:170–CG:171
broadcasting commands, CG:175–CG:176
escalating messages, CG:177
evaluating action results, CG:164
overview, CG:160–CG:161
owning messages, CG:162–CG:163
process, CG:128–CG:129
reading operator instructions,
CG:168–CG:169
verifying
automatic actions, CG:165–CG:166
operator-initiated actions, CG:167
sources, message correlation, CG:429
Spanish
OVO, AR:307
SPARCclassic. See Sun SPARCclassic
SPARCserver. See Sun SPARCserver
SPARCstation. See Sun SPARCstation
special characters, flexible management
templates, AR:124
SSH
OVO agent installation, AR:57–AR:61
security, AR:461
SSP
configuring, DCE:307–DCE:308
establishing remote host equivalence,
DCE:308
exporting SSP logfiles directory, DCE:308
reconfiguring
snmpd daemon, DCE:307
SSP templates, DCE:309, DCE:310
SSP Tools, DCE:306
SSP Config application, DCE:306
SSP message group, AR:73
standard de-installation
See also de-installing
OVO
MPE/iX, DCE:163
SINIX RM/Reliant, DCE:262
Solaris, DCE:285
Tru64 UNIX, DCE:334
Windows NT/2000, DCE:385

OVPA
HP-UX, AR:216
Solaris, AR:216

standard installation
See also installing
OVO
HP-UX, DCE:85
Linux, DCE:137
MPE/iX, DCE:163
SINIX RM/Reliant, DCE:261
Solaris, DCE:280
Windows NT/2000, DCE:373–DCE:378

OVPA
HP-UX, AR:212
Solaris, AR:212

standard scenarios, automating, CG:364
Start Customized Application wizard figures
  broadcasting commands, CG:176
  Step 2 of 3, CG:171
  Step 3 of 3, CG:174
Start extract application, AR:218
Start Perf Agt application, AR:219
Start pv application, AR:219
Start pvalarmd application, AR:219
Start Services application, DCE:420
Start utility application, AR:219

starting
applications, CG:170
accounts, AR:466
  managed nodes, AR:261–AR:262
  remotely, AR:467
broadcast commands
  managed nodes, AR:261–AR:262
  remotely, AR:467
corrective actions, CG:393
EMS GUI, DCE:116, DCE:117
I/O applications remotely, AR:467
operator-initiated actions, CG:167
OVO from operator GUI, CG:222
SMIT User Interface, DCE:57

streamed jobs on MPE/iX managed nodes,
DCE:161
startup options, Java GUI, AR:321–AR:322
state-based browsers, CG:364,
  CG:411–CG:412
statistical reports, AR:115
status
  application, HTTPS:185
status bar
  figure, CG:105
overview, CG:104
Status Propagation display mode, CG:163,
  CG:293
status variables, AR:133
status.alarmgen logfile template, AR:220
status.mi logfile template, AR:220
status.perflbd logfile template, AR:220
status.pv logfile template, AR:222
status.pvalarmd logfile template, AR:222
status.rep_server logfile template, AR:220
status.scope logfile template, AR:220
status.tfd logfile template, AR:221
stderr action, CG:164
stdout and stderr devices, reporting errors,
AR:383
stderr action, CG:164
Stop Perf Agt application, AR:219
Stop pvalarmd application, AR:219
Stop Services application, DCE:421

strategies
  message filtering, CG:355–CG:378
  message forwarding, CG:480–CG:482
streamed jobs on MPE/iX managed nodes
  customizing job stream facility, DCE:161
overview, DCE:161–DCE:163
preparing OVO, DCE:163
starting, DCE:161
SYSSTART.PUB.SYS parameters, DCE:161
strings, time zone, AR:135–AR:136
subproduct option, AR:326
subproducts
  English, DCE:83
  sub-tree on management server
    customer-specific, DCE:81
    vendor-specific, DCE:80
sudo
  setting up, HTTPS:82
  working with, HTTPS:81
Master Index

Sun Clusters
See also Solaris managed nodes; Sun Enterprise E10000 support, DCE:312

Sun Enterprise E10000
See also Solaris managed nodes; Sun Clusters
installing OVO agent, DCE:309–DCE:310
logfile templates, DCE:304
managing, DCE:301–DCE:302
monitored objects, DCE:305
monitoring, DCE:301–DCE:302
operating system versions, DCE:302
overview, DCE:301–DCE:310
preconfigured elements, DCE:302–DCE:306
SNMP trap interception, DCE:303
SSP
configuring, DCE:307–DCE:308
SSP Tools, DCE:306
template groups, DCE:302

Sun Management Center, DCE:311
See also Solaris managed nodes
Sun Microsystems. See Solaris managed nodes; Sun Clusters; Sun Enterprise E10000; Sun Management Center; Sun SPARCclassic; Sun SPARCserver; Sun SPARCstation; Sun Ultra

Sun Solaris. See Solaris
Sun SPARCclassic
See also Solaris managed nodes
Sun SPARCserver
See also Solaris managed nodes
Sun SPARCstation, DCE:294
See also Solaris managed nodes
Sun Ultra
See also Solaris managed nodes
supported platforms, HTTPS:28
suppress
See also suppressing; suppression conditions
deploying, CG:356
description, CG:334–CG:337
types, verifying, CG:371–CG:373
SUPPRESS parameter, AR:131
suppressing
See also suppress; suppression
duplicate messages, CG:370
flexible management environments,
management server, CG:376
multiple messages, CG:329
unmatched conditions, CG:356
suppression
See also suppress; suppressing
counter, CG:375
time, CG:374
Switch User template, CG:438
switching
backup server, CG:469
message control, CG:473–CG:474
primary management responsibility,
CG:467–CG:468
reversing switch, CG:468
switching message colors to entire line,
CG:215
symptoms, analyzing, AR:379
synchronizing
commands with OVO agent character set,
AR:286
OVO and NNM event correlation, CG:431
syntax
EMS Resources application, DCE:119
opclic command, AR:512
pattern-matching, CG:341–CG:343
proxies, HTTPS:142
templates
flexible management, AR:124–AR:129
management responsibility switching,
AR:126
message operations and target rules,
AR:127
responsible manager configuration,
AR:125
scheduled outages, AR:128, AR:130
service hours, AR:128, AR:130
time, AR:126
time zone strings, AR:135
SYSSTART.PUB.SYS parameters, DCE:161
System Administrator. See SAM
System Log (MetaFrame) template, DCE:437
System Log (Terminal Server) template,
DCE:437
system resource files
AIX, DCE:61
HP-UX, DCE:109
MPE/iX, DCE:178
Novell NetWare, DCE:218
Sequent DYNIX, DCE:236
SGI IRIX, DCE:250
SINIX RM/Reliant, DCE:268
Solaris, DCE:296
Tru64 UNIX/T, DCE:349
Windows NT/2000, DCE:431

system security
exception warnings, AR:343
System Summary application, DCE:214

T
<$T>$ variable, AR:168
tables and tablespaces
non-OVO, AR:552
OVO, AR:547

tabs, adding to browser pane, CG:214
Tail Status Files application, AR:219
tailored set of applications, CG:207
tailored_applications_start option, AR:327
target directories
See also directories; temporary directories
AIX, DCE:59
HP-UX, DCE:103
Linux, DCE:146
MPE/iX, DCE:176
Novell NetWare, DCE:216
SGI IRIX, DCE:247
SINIX RM/Reliant, DCE:265
Solaris, DCE:294
Tru64 UNIX/T, DCE:338
Windows NT/2000, DCE:428
target rules, messages, CG:465–CG:466
tasks
OVO, CG:44–CG:54
TCP/IP
tools, HTTPS:188
TCP/IP Status application, DCE:422
techniques, C2 security, CG:226
template administrators
See also operators; templates; users; OVO administrator
description, CG:58
template conditions, CG:46
See also templates
Template Detail Report, AR:111
template groups
See also templates
advantages, CG:310
creating, CG:311
hierarchies
creating, CG:311
navigating, CG:311
organizing, CG:310–CG:311
preconfigured
HP-UX (OVO), DCE:97
HP-UX (OVPA), AR:220–AR:222
Linux, DCE:142
Solaris (OVO), DCE:287
Solaris (OVPA), AR:220–AR:222
Sun Enterprise E10000, DCE:302
templates
See also template administrators; template conditions; template groups
adding
new combination of nodes and templates, CG:314
SNMP traps, CG:418
assigning, CG:313–CG:315
configuring
application-specific, CG:329
multiple, CG:326
creating for message sources, CG:309
distributing
assigned, CG:315
description, CG:305
message source, CG:307–CG:316
EMS
configuring, DCE:120
event correlation example, CG:435–CG:438
flexible management
configuring, AR:117–AR:153
examples, AR:146–AR:153
follow-the-sun responsibility switch, AR:148–AR:149
keywords, AR:119–AR:123
location, AR:117
message forwarding between
management servers, AR:150–AR:151
responsibility switch, AR:146–AR:147
scheduled outages, AR:153
service hours, AR:152
Master Index

syntax, AR:124–AR:129
types, AR:117
generic, CG:329
importing OVO templates into OpenView
  Operations for Windows, AR:241
logfile, CG:385
  Citrix MetaFrame, DCE:437
  HP-UX (OVO), DCE:98
  Linux, DCE:142
  SGI IRIX, DCE:245
  Solaris (OVO), DCE:288
  Sun Enterprise E10000, DCE:304
  Tru64 UNIX, DCE:335
variables, AR:162
management responsibility switching,
  AR:126
message
  HP-UX (OVO), DCE:97
  Solaris (OVO), DCE:287
message forwarding, CG:476–CG:477
  attributes, AR:138
  configuring, AR:138
  location, AR:137
  parameters, AR:139
  troubleshooting, CG:483
message operations syntax, AR:127
message source variables, AR:155–AR:169
message target rule syntax, AR:127
MPE/iX console messages
  default attributes, CG:424
  defining, CG:423
multiple, CG:327–CG:328
protecting distribution, AR:470
responsible manager, CG:464
scheduled outage syntax, AR:128–AR:130
scheduling, AR:130–AR:136
service hours
  location, AR:130
  parameters, AR:131
  syntax, AR:128, AR:130
SNMP trap variables, AR:165–AR:168
SSP, reconfiguring, DCE:309, DCE:310
Switch User, CG:438
threshold monitor
  EMS, DCE:113
  variables, AR:163
time, CG:466
  examples, AR:141–AR:143
  keywords, AR:144–AR:145
  overview, AR:140–AR:145
  syntax, AR:126
time-indifferent, CG:466
  Transient Interface Down, CG:437
  Transient Node Down, CG:436
Templates Groups list box, CG:310
Templates Overview Report, AR:112
Templates Summary Report, AR:112
temporary directories
  See also directories; target directories
  AIX, DCE:59
  HP-UX, DCE:103
  Linux, DCE:146
  MPE/iX, DCE:176
  Novell NetWare, DCE:216
  Sequent DYNIX, DCE:233
  SGI IRIX, DCE:247
  SINIX RM/Reliant, DCE:265
  Solaris, DCE:294
  Tru64 UNIX, DCE:338
  Windows NT/2000, DCE:428
temporary files, excluding from automatic
  backups, AR:491
terminal access, CG:177, CG:226
text, reviewing original message, CG:146
<$THRESHOLD> variable, AR:163
threshold monitors
  conditions
    advanced monitoring, CG:409–CG:410
    examples, CG:413
    multiple, CG:411–CG:412
    configuring, CG:408
    default, CG:409
    integrating, CG:406–CG:409
  messages, CG:395–CG:413
  templates
    EMS, DCE:113
    variables, AR:163
  thresholds
    maximum, CG:401
    minimum, CG:401
ticket, RPC server, AR:457
time
  attributes, CG:449
configuring time-indifferent templates, CG:466
setting intervals, CG:466
templates
description, CG:466
examples, AR:141–AR:143
keywords, AR:144–AR:145
overview, AR:140–AR:145
syntax, AR:126
zone, AR:135
time-based suppression, CG:374
Time message attribute, AR:77
timeouts, configuring for report generation, AR:109
Tips_for_Installing_Agents, DCE:135
title_suffix option
ito_op, AR:322
itooprc, AR:327
To De-install an Agent Manually, DCE:140
toolbar
figure, CG:107
overview, CG:107
tools
backup, AR:488
controller, AR:333–AR:334
license maintenance, AR:512–AR:513
node mapping, AR:334–AR:335
recovery, AR:488
tour, Java GUI, CG:65–CG:66
trace (ASCII) file, AR:359
trace option
ito_op, AR:322
itooprc, AR:327
tracing
commands, AR:67
events, AR:67
problems, AR:378
Transient Interface Down template, CG:437
Transient Node Down template, CG:436
traps
SNMP, CG:414–CG:421
well-defined, DCE:99
trend-analysis reports, AR:115
Trend Parameters* application, DCE:214
trouble ticket services
forwarding messages, AR:133
trouble ticket system
concepts, AR:265
configuring, AR:269
connecting management servers, CG:480
parameters, AR:270
writing scripts and programs,
AR:266–AR:267
troubleshooting, HTTPS:184
AIX managed nodes, DCE:49
application status, HTTPS:185
authentication, HTTPS:199
certificate deployment, HTTPS:208
certificates, HTTPS:199
communication, HTTPS:190, HTTPS:192
database, AR:385–AR:387
embedded performance component,
AR:428–AR:432
installed OV filesets, HTTPS:186
basic inventory, HTTPS:186
detailed inventory, HTTPS:187
native inventory, HTTPS:187
IP aliases, DCE:49
logging, HTTPS:189
managed node runtime, AR:401–AR:415
management server
GUI, AR:390–AR:392
message forwarding template, CG:483
OVO, AR:388–AR:389
MPE/iX managed nodes
installation, AR:395–AR:398
runtime, AR:420–AR:426
multi-homed host installation,
AR:435–AR:442
network, HTTPS:190
NSF, AR:443
OvCoreId, HTTPS:209
overview, AR:375–AR:384
OVO communication, HTTPS:204
OVO in a Cluster environment,
DCE:459–DCE:462, ??–DCE:462
ping applications, HTTPS:184
PRC daemons or local location brokers,
AR:427
registered applications, HTTPS:185
RPC call, HTTPS:188
TCP/IP tools, HTTPS:188
tools, HTTPS:184
UNIX managed nodes
installation, AR:393
runtime, AR:416–AR:419
what string, HTTPS:186
Windows managed nodes
installation, AR:399–AR:400
Tru64 UNIX managed nodes
DCE
configuring, DCE:326–DCE:327
removing, DCE:327
OVO
default operator, DCE:348
directory structure, DCE:347
file locations, DCE:347
hardware requirements, DCE:317
include file, DCE:351
installation requirements,
DCE:317–DCE:320
installation tips, DCE:323–DCE:325
libraries, DCE:350–DCE:352
logfile templates, DCE:335
makefile, DCE:352
organization, DCE:347–DCE:349
overview, DCE:315–DCE:353
preconfigured elements,
DCE:335–DCE:336
scripts and programs, DCE:337–DCE:338
SNMP event interceptor (not supported), DCE:335
software requirements, DCE:318–DCE:320
system resource files, DCE:349
OVPA, AR:207
trusted system security. See C2 security
TS_Licensing object, DCE:436
TS_Service object, DCE:436
ttd monitor template, AR:221
ttnsrp pipe file, AR:354
ttnsrq queue file, AR:354
ttnsp pipe file, AR:354
ttnsq queue file, AR:354
tuning performance, AR:370–AR:374
Types of Default Applications, DCE:56

U
U message attribute, AR:75
Ultra. See Sun Ultra
unbuffering messages
automatically, CG:439
manually, CG:439–CG:440
unbuffering pending messages, CG:99
UNIX
distribution tips, AR:194
distribution parameters, AR:194
managed nodes
assigning passwords, AR:469
troubleshooting
installation, AR:393
runtime, AR:416–AR:419
Unknown message severity level, AR:74
unknown nodes
select all, HTTPS:162
unmatched
conditions, suppressing, CG:356
messages, classifying, CG:49
unmatched
messages, forwarding, AR:382
Unmonitored Report, AR:112
update
root certificate, HTTPS:54
updating current workspace, CG:86–CG:88
updating OVO on managed nodes
agents, AR:48–AR:56
configuration, AR:187–AR:203
uploading configuration files, CG:470
URL Shortcuts folder
figures
object tree, CG:78
starting application, CG:87
updating application, CG:88
overview, CG:78
Used Shares application, DCE:423
User Action Report, AR:112
User Audit Report, AR:112
User Logon Report, AR:112
user option
ito_op, AR:322
itooprc, AR:327
User Profile Overview Report, AR:112
User Profile Report, AR:112
<$USER$> variable, AR:169
users
See also operators; template
administrators; OVO administrator
changing
names, AR:462
passwords, AR:462

296

<table>
<thead>
<tr>
<th>AR</th>
<th>OVO Administrator’s Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>DCE</td>
<td>OVO DCE Agent Concepts and Configuration Guide</td>
</tr>
<tr>
<td>CG</td>
<td>OVO Concepts Guide</td>
</tr>
<tr>
<td>HTTPS</td>
<td>OVO HTTPS Agent Concepts and Configuration Guide</td>
</tr>
</tbody>
</table>
Master Index

<table>
<thead>
<tr>
<th>Concept/Action</th>
<th>Page(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>concept, CG:55–CG:61</td>
<td></td>
</tr>
<tr>
<td>controlling passwords, AR:462</td>
<td></td>
</tr>
<tr>
<td>logged into Java GUI, AR:343</td>
<td></td>
</tr>
<tr>
<td>profiles, CG:56</td>
<td></td>
</tr>
<tr>
<td>roles, CG:55</td>
<td></td>
</tr>
<tr>
<td>root, AR:466</td>
<td></td>
</tr>
<tr>
<td>Users application, DCE:214, DCE:441</td>
<td></td>
</tr>
<tr>
<td>V</td>
<td></td>
</tr>
<tr>
<td>&lt;$V$&gt; variable, AR:168</td>
<td></td>
</tr>
<tr>
<td>&lt;$VALAVG$&gt; variable, AR:163</td>
<td></td>
</tr>
<tr>
<td>&lt;$VALCNT$&gt; variable, AR:163</td>
<td></td>
</tr>
<tr>
<td>&lt;$VALUE$&gt; variable, AR:163</td>
<td></td>
</tr>
<tr>
<td>variables</td>
<td></td>
</tr>
<tr>
<td>See also parameters</td>
<td></td>
</tr>
<tr>
<td>action, AR:160–AR:161</td>
<td></td>
</tr>
<tr>
<td>adding OVO, CG:174</td>
<td></td>
</tr>
<tr>
<td>applications, AR:171–AR:186</td>
<td></td>
</tr>
<tr>
<td>environmental, AR:155</td>
<td></td>
</tr>
<tr>
<td>GUI, AR:171–AR:186</td>
<td></td>
</tr>
<tr>
<td>language, AR:277</td>
<td></td>
</tr>
<tr>
<td>instruction text interface, AR:170</td>
<td></td>
</tr>
<tr>
<td>message source templates, AR:155–AR:169</td>
<td></td>
</tr>
<tr>
<td>messages</td>
<td></td>
</tr>
<tr>
<td>MPE/iX console, AR:164</td>
<td></td>
</tr>
<tr>
<td>scheduled actions, AR:169</td>
<td></td>
</tr>
<tr>
<td>monitoring, CG:401</td>
<td></td>
</tr>
<tr>
<td>opcinfo, HTTPS:126</td>
<td></td>
</tr>
<tr>
<td>opcsvinfo, HTTPS:126</td>
<td></td>
</tr>
<tr>
<td>overview, AR:154–AR:186</td>
<td></td>
</tr>
<tr>
<td>resolving, AR:159</td>
<td></td>
</tr>
<tr>
<td>setting, HTTPS:126</td>
<td></td>
</tr>
<tr>
<td>status, AR:133</td>
<td></td>
</tr>
<tr>
<td>templates</td>
<td></td>
</tr>
<tr>
<td>logfile, AR:162</td>
<td></td>
</tr>
<tr>
<td>SNMP trap, AR:165–AR:168</td>
<td></td>
</tr>
<tr>
<td>threshold monitor, AR:163</td>
<td></td>
</tr>
<tr>
<td>types, AR:154</td>
<td></td>
</tr>
<tr>
<td>vendor-specific sub-tree on management server, DCE:80</td>
<td></td>
</tr>
<tr>
<td>verifying</td>
<td></td>
</tr>
<tr>
<td>automatic actions, CG:165–CG:166</td>
<td></td>
</tr>
<tr>
<td>operator-initiated actions, CG:167</td>
<td></td>
</tr>
<tr>
<td>RPC server ticket, AR:457</td>
<td></td>
</tr>
<tr>
<td>suppress types, CG:371–CG:373</td>
<td></td>
</tr>
<tr>
<td>versions</td>
<td></td>
</tr>
<tr>
<td>OVO, AR:376–AR:377</td>
<td></td>
</tr>
<tr>
<td>OVO agent</td>
<td></td>
</tr>
<tr>
<td>displaying available, AR:65</td>
<td></td>
</tr>
<tr>
<td>displaying installed, AR:65</td>
<td></td>
</tr>
<tr>
<td>managing, AR:64</td>
<td></td>
</tr>
<tr>
<td>removing, AR:66</td>
<td></td>
</tr>
<tr>
<td>programs, AR:190</td>
<td></td>
</tr>
<tr>
<td>scripts, AR:190</td>
<td></td>
</tr>
<tr>
<td>viewing</td>
<td></td>
</tr>
<tr>
<td>EMS GUI resource instances, DCE:116</td>
<td></td>
</tr>
<tr>
<td>overview, CG:151–CG:155</td>
<td></td>
</tr>
<tr>
<td>messages</td>
<td></td>
</tr>
<tr>
<td>in message browser, CG:133</td>
<td></td>
</tr>
<tr>
<td>OVPA documentation, AR:223</td>
<td></td>
</tr>
<tr>
<td>virtual node, HTTPS:146</td>
<td></td>
</tr>
<tr>
<td>adding, HTTPS:147</td>
<td></td>
</tr>
<tr>
<td>assigning policies, HTTPS:150</td>
<td></td>
</tr>
<tr>
<td>cluster, HTTPS:146</td>
<td></td>
</tr>
<tr>
<td>de-assigning policies, HTTPS:150</td>
<td></td>
</tr>
<tr>
<td>deleting, HTTPS:149</td>
<td></td>
</tr>
<tr>
<td>deploying policies, HTTPS:151</td>
<td></td>
</tr>
<tr>
<td>modifying, HTTPS:149</td>
<td></td>
</tr>
<tr>
<td>modifying policies, HTTPS:151</td>
<td></td>
</tr>
<tr>
<td>physical node, HTTPS:146</td>
<td></td>
</tr>
<tr>
<td>Virtual Terminal application, DCE:174, DCE:176</td>
<td></td>
</tr>
<tr>
<td>Virtual Terminal PC application, DCE:424</td>
<td></td>
</tr>
<tr>
<td>Volume application, DCE:214</td>
<td></td>
</tr>
<tr>
<td>vt3k operation, DCE:179</td>
<td></td>
</tr>
<tr>
<td>W</td>
<td></td>
</tr>
<tr>
<td>Warning message severity level, AR:74</td>
<td></td>
</tr>
<tr>
<td>web browser</td>
<td></td>
</tr>
<tr>
<td>choosing, CG:204</td>
<td></td>
</tr>
<tr>
<td>figures</td>
<td></td>
</tr>
<tr>
<td>embedded web browser, CG:102</td>
<td></td>
</tr>
<tr>
<td>proxy settings, CG:103</td>
<td></td>
</tr>
<tr>
<td>overview, CG:100–CG:103</td>
<td></td>
</tr>
<tr>
<td>web reporting, restricting, AR:116</td>
<td></td>
</tr>
<tr>
<td>web_browser_type option, AR:327</td>
<td></td>
</tr>
<tr>
<td>well-defined traps, DCE:99</td>
<td></td>
</tr>
<tr>
<td>what string, HTTPS:186</td>
<td></td>
</tr>
<tr>
<td>which_browser option, AR:327</td>
<td></td>
</tr>
<tr>
<td>windows</td>
<td></td>
</tr>
<tr>
<td>managed node</td>
<td></td>
</tr>
<tr>
<td>Add Node for External Events, CG:236</td>
<td></td>
</tr>
</tbody>
</table>

AR  OVO Administrator's Reference
DCE OVO DCE Agent Concepts and Configuration Guide
CG  OVO Concepts Guide
HTTPS OVO HTTPS Agent Concepts and Configuration Guide
Node Advanced Options, CG:244
Node Communication Options, CG:245
OVO Add Node, CG:243
OVO Add Nodes, CG:236
OVO Node Bank, CG:229–CG:230
OVO Node Hierarchy Bank, CG:231–CG:235
primary windows, CG:228
NetWare
NetWare Config, DCE:206
NetWare Performance, DCE:207–DCE:208
NetWare Tools, DCE:208
operator
Application Desktop, CG:60
Managed Nodes, CG:60
Message Browser, CG:61
Message Groups, CG:60
OVO administrator
Configure Management Server, AR:193
Download Configuration Data, AR:486–AR:487
Install/Update OVO Software and Configuration, AR:51, AR:189
Message Group Bank, AR:72
Node Group Bank, AR:71
template administrator
Add Configuration window, CG:314
Add MPE/iX Console Messages, CG:423
Add SNMP Trap, CG:418
Condition No., CG:410
Define Configuration, CG:313
Message and Suppress Conditions, CG:337
Message Condition Advanced Options, CG:418
Message Correlation, CG:360
Message Source Template, CG:309
Message Source Templates, CG:316
Modify OVO Interface Messages, CG:392
Regroup Conditions, CG:382
Windows Installation Server requirements, DCE:358
Windows managed nodes troubleshooting
installation, AR:399–AR:400
Windows managed nodes requirements, DCE:358
Windows NT/2000 managed nodes
agent accounts, DCE:364–DCE:366
alternative accounts, DCE:365–DCE:366
applications, DCE:394–DCE:426
assigning passwords, AR:470
Citrix MetaFrame
applications, DCE:438–DCE:441
integration, DCE:433–DCE:437
default operator, DCE:430
de-installing agents, DCE:385
directory structure, DCE:429
file locations, DCE:430
FTP
installing agents, DCE:367–DCE:372
re-installing agents, DCE:378–DCE:381
hardware requirements, DCE:357–DCE:358
HP ITO Account, DCE:364
include file, DCE:432
installation
methods, DCE:363
requirements, DCE:357–DCE:360
installing agents, DCE:361–DCE:384
libraries, DCE:432
logfile locations, AR:508
makefile, DCE:432
management server requirements, DCE:357
node requirements, DCE:358
organization, DCE:429–DCE:431
overview, DCE:355–DCE:448
preconfigured elements, DCE:386–DCE:393
pre-installing agents, DCE:382–DCE:384
re-installing agents, DCE:378–DCE:381
scripts and programs, DCE:427–DCE:428
SMS integration, DCE:442–DCE:447
SNMP event interceptor, DCE:388–DCE:391
software requirements, DCE:359–DCE:360
system resources, DCE:431
Windows Installation Server requirements, DCE:358
WMI policy, changing default name, AR:240
Working OVO Operators Report, AR:112
workspace pane
accessing OpenView applications, CG:156
evaluating action results, CG:164
figures
graphs and charts, CG:81
main window, CG:79
message browser, CG:91
298
AR OVO Administrator's Reference
DCE OVO DCE Agent Concepts and Configuration Guide
CG OVO Concepts Guide
HTTPS OVO HTTPS Agent Concepts and Configuration Guide
moving (after), CG:200
moving (before), CG:199
popup menu on pane, CG:114
popup menu on tab, CG:113
finding impacted Service Navigator services, CG:156
investigating problems, CG:150
moving, CG:199
overview, CG:79–CG:81
popup menus, CG:113
workspaces
  Corrective Actions, CG:84
  Diagnostic Dashboard, CG:83
  Message Dashboard, CG:82
  Online Help, CG:85
  Services, CG:82
  updating current, CG:86–CG:88
Workspace Properties dialog box figure, CG:102
workspaces, assigned by the OVO administrator, CG:193
Workst Stats application, DCE:426
worldwide management. See follow-the-sun control
worldwide management domain, CG:448
writing to default working directory, AR:463

X
X resources
  fonts, AR:279–AR:283
<$X$> variable, AR:168
<$x$> variable, AR:168
XCONSOLE application, DCE:214
X-OVw group applications, AR:330

Z
zone, time
  parameter, AR:136
string, AR:135
## Master Index

<table>
<thead>
<tr>
<th>300</th>
<th>AR</th>
<th>OVO Administrator’s Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>DCE</td>
<td>OVO DCE Agent Concepts and Configuration Guide</td>
</tr>
<tr>
<td></td>
<td>CG</td>
<td>OVO Concepts Guide</td>
</tr>
<tr>
<td></td>
<td>HTTPS</td>
<td>OVO HTTPS Agent Concepts and Configuration Guide</td>
</tr>
</tbody>
</table>