
# Opsware® SAS 5 Deployment and Installation Guide

**Corporate Headquarters**

599 North Mathilda Avenue Sunnyvale, California 94085 U.S.A.
T + 1 408.744.7300 F +1 408.744.7383 www.opsware.com

Opsware SAS Version 5.5.1

# Table of Contents

# Preface

Welcome to the Opsware Server Automation System (SAS) – an enterprise-class software solution that enables customers to get all the benefits of the Opsware data center automation platform and support services. Opsware SAS provides a core foundation for automating formerly manual tasks associated with the deployment, support, and growth of server and server application infrastructure.

## Overview of this Guide

This guide describes how to use the Opsware Installer to install the software components that make up an Opsware core. It also describes the administrative tasks required prior to installing an Opsware core.

This guide is intended for Unix system administrators, database administrators, and network administrators.

## Contents of this Guide

This guide contains the following chapters:

**Chapter 1: Opsware System Overview and Checklist**: Describes the types of Opsware SAS core installations.

**Chapter 2: Pre-installation Requirements**: Describes the system and network administration tasks that must be performed before you can run the Opsware Installer. It also provides information about the Opsware SAS requirements for installing the Oracle database.

**Chapter 3: Prerequisite Information for the Installer Interviewer**: Lists the information you will be prompted for by the Opsware Installer interviewer. This chapter also provides information about the installer command line syntax, log files, and how the Opsware Installer is distributed on DVD and across CD-ROMs.

**Chapter 4: Opsware Standalone Installation**: Describes how to run the Opsware Installer to create a standalone core.

**Chapter 5: Post-Installation Tasks**: Describes system administration tasks that you must perform after installing a core.

**Chapter 6: Opsware Multimaster Installation**: Describes how to run the Opsware Installer to upgrade a standalone core to multimaster and install target facilities.

**Chapter 7: Opsware Satellite Installation**: Describes how to run the Opsware Installer for creating an Opsware satellite realm.

**Chapter 8: What's Next**: Provides an overview of the configuration tasks required for the Opsware SAS after the core has been installed.

**Chapter 9: Opsware Core Uninstallation**: Shows how to un-install a standalone core, remove a core from a multimaster mesh, and un-install an entire Opsware SAS made up of multiple cores in different facilities.

## Conventions in this Guide

This guide uses the following typographical and formatting conventions.

| NOTATION | DESCRIPTION |
|---|---|
| **Bold** | Identifies field menu names, menu items, button names, and inline terms that begin with a bullet. |
| `Courier` | Identifies text that is entered or displayed at the command-line prompt, such as Unix commands, Opsware SAS commands, file names, paths, directories, environment variable names, contents of text files that are viewed or edited with a text editor, source code in a programming language, and SQL (database) commands. |
| *Italics* | Identifies document titles, DVD titles, web site addresses. Used to introduce new terms when they are first defined in a document and for emphasis. |

## Icons in this Guide

This guide uses the following iconographic conventions.

| ICON | DESCRIPTION |
|---|---|
|  | This icon represents a note. It identifies especially important concepts that warrant added emphasis. |
|  | This icon represents a requirement. It identifies a task that must be performed before an action under discussion can be performed. |
|  | This icon represents a tip. It identifies information that can help simplify or clarify tasks. |
|  | This icon represents a warning. It is used to identify significant information that must be read before proceeding. |

## Guides in the Documentation Set and Associated Users

- The *Opsware® SAS User's Guide* is intended to be read by the system administrator who is responsible for performing the day-to-day functions of managing servers, provisioning operating systems, uploading packages, setting up the Software Tree and node hierarchies, attaching software applications and installing them on servers, managing patches, reconciling servers with software, creating and executing scripts, tracking configuration, and deploying and rolling back code and content. It also documents the day-to-day functions of managing servers, such as server compliance and auditing, software packaging, application configuration, agent deployment, and global shell remote data center management.
- The *Opsware® SAS Administration Guide* is intended to be read by Opsware administrators who will be responsible for monitoring and diagnosing the health of the Opsware SAS components.
- The *Opsware® SAS Deployment and Installation Guide* is intended to be used by system administrators who are responsible for the installation of Opsware SAS in a facility. It documents how to run the Opsware Installer and how to configure each of the components.
- The *Planning Deployments for Opsware® SAS* is intended to be used by advanced system administrators who will be responsible for planning all facets of an Opsware SAS installation and deployment. It documents all the main features of Opsware SAS and scopes out the planning tasks necessary to successfully deploy Opsware SAS. Sections include: planning the Opsware SAS design for a core, types of installations, and discusses business goals that can be achieved using the software. It also includes information on system sizing, checklists, and best practices.
- The *Opsware® SAS Configuration Guide* is intended to be used by system administrators who are responsible for all facets of configuring the Opsware Command Center. It documents how to set up users and groups, how to configure Opsware server management, and how to set up the main Opsware Command Center features, such as patch management, configuration tracking, software repository replicator setup, code deployment, and software provisioning.

## Opsware, Inc. Contact Information

The main web site and phone number for Opsware, Inc. are as follows:

- *http://www.opsware.com/index.htm*
- +1 (408) 744-7300

For links to the latest product documentation and software downloads, see the Opsware Customer Support site:

- *https://download.opsware.com/opsw/main.htm*

For troubleshooting information, you can search the Opsware Knowledge Base at:

- *https://download.opsware.com/kb/kbindex.jspa*

The Opsware Customer Support email address and phone number follow:

- support@opsware.com
- +1 (877) 677-9273

# Chapter 1: Installation Overview and Checklists

**IN THIS CHAPTER**

This section discusses the following topics:

- Types of Opsware SAS Installations
- Opsware Core Installation Process Flow
- Changes to the Installation Process Since Version 4
- Checklists

## Types of Opsware SAS Installations

There are three basic types of Opsware SAS installations: standalone, multimaster, and satellite.

- **Standalone**: A standalone core does not communicate or exchange information with other cores. A standalone core manages servers in a single facility. (Optionally, a standalone core can also manage servers in remote facilities installed with Opsware Satellites.) A core contains all components of Opsware SAS, except for the Opsware Agents, which run on the servers managed by the core.
- **Multimaster**: A multimaster core exchanges information with other cores. This collection of cores is called a multimaster mesh. With a multimaster mesh, you can centralize the management of several facilities but still get the performance benefits of having a local copy of key Opsware SAS data at each facility.
- **Satellite**: Installed in a remote facility, an Opsware Satellite provides network connection and bandwidth management for a core that manages remote servers. A Satellite must be linked to at least one core, which may be either standalone or multimaster.

This guide uses the term facility to refer to the collection of servers and devices that reside in a single physical location. A facility can be all or part of a data center, server room, or computer lab. Each Opsware core or Satellite is associated with a specific facility.

## Opsware Core Installation Process Flow

Figure 1-1 shows the overall process of an Opsware core installation. The main phases of the installation process follow:

1. **Planning**: Decide which type of Opsware SAS installation is appropriate for your site and what hardware needs to be procured. At the end of this phase, you may follow the instructions in this installation guide.

   See the *Planning Deployments for Opsware® SAS*.

2. **Pre-installation Requirements**: At this point, you have the necessary hardware in place and you are ready to install an Opsware core. In this phase you perform hands-on administrative tasks such as resolving host names, opening ports, and installing the necessary OS utilities or patches.

   See Chapter 2, “Pre-Installation Requirements” on page 29 of this guide.

3. **Pre-requisite Info for Installer Interview**: Gather information for the Opsware Installer interview, which prompts you for information about the core and your operational environment. This information includes the name of the facility to be managed by the core, the authorization domain, as well as information about the Oracle database that underlies the Opsware Model Repository.

   At the end of this phase, you are ready to run the Opsware Installer to perform one of the following three types of installations.

   See Chapter 3, “Prerequisite Information for the Installer Interview” on page 41 of this guide.

4. **Standalone Core Installation**: Run the Opsware Installer for the interview and then create the core.

   See Chapter 4, “Opsware Standalone Installation” on page 65 of this guide.

   Or

5. **Multimaster Core Installation**: Run the Opsware Installer for the interview and then add a core to a multimaster mesh.

See Chapter 6, “Opsware Multimaster Installation” on page 91 of this guide.

Or

6 **Satellite Realm Installation**: Run the Opsware Installer for the interview and create an Opsware Satellite in a remote facility.

See Chapter 7, “Opsware Satellite Installation” on page 109 of this guide.

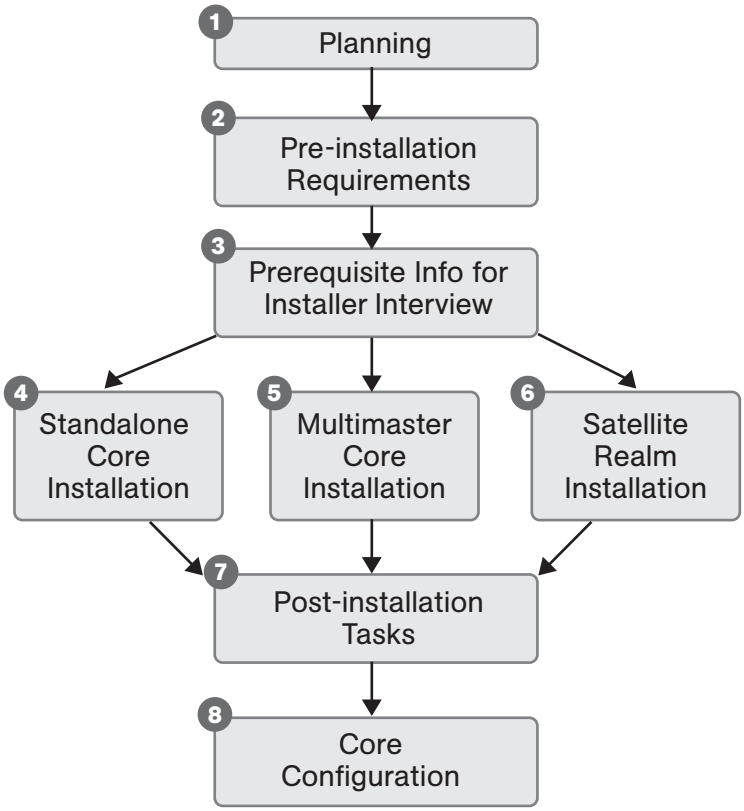
7 **Post-installation Tasks**: Perform hands-on administrative tasks such as configuring the DHCP server in preparation for Opsware OS Provisioning. At the end of this phase, the newly installed Opsware core is up and running.

See Chapter 5, “Post-Installation Tasks” on page 73 of this guide.

8 **Core Configuration**: Configure Opsware SAS, performing tasks such as creating Opsware users, groups, and the software tree. At the end of this phase, Opsware SAS is ready for operational use by system administrators.

See the *Opsware® SAS Configuration Guide*.

*Figure 1-1: Opsware Core Installation Process Flow*




## Changes to the Installation Process Since Version 4

Since version 4 of Opsware SAS, the installation process has changed in the following ways:

- The Oracle database (Model Repository) requires the following new tablespaces:

  ```
  AAA_DATA
  AAA_INDX
  ```

- In the Opsware Installer, the selection menu displays the components in the order they must be installed.
- The Opsware Installer includes the new core components: Opsware Gateway and Opsware Global File System.
- With each invocation of the Opsware Installer, you can install multiple components.

- The Opsware Installer can install an Opsware Satellite.
- The configuration of TIBCO Rendezvous (required only for multimaster) is automatic.
- The Opsware Installer has two modes: simple and advanced. The simple mode uses default values for some parameters. The advanced mode prompts for all parameters.
- In a multimaster installation, the Opsware Installer defines the new facility and the multimaster central Data Access Engine. (In version 4, you performed these tasks with the Opsware Command Center.)
- After installing the core, perform the following tasks:
  - On a Windows server, install the Windows Agent Deployment Helper, which is required for the Opsware Discovery and Deployment (ODAD) feature.
  - On a Windows server, install the ISM Development Kit (IDK), which is required for the Opsware Visual Packager.

## Checklists

This section discusses the following topics:

- Overall Planning Checklist
- Specific Core Planning Checklist
- Specific Core Requirements Checklist
- Pre-Installation Tasks Checklist
- Post-Installation Tasks Checklist

### Overall Planning Checklist

The following checklist summarizes decisions regarding the overall design of your Opsware SAS installation.

*Table 1-1: Overall Planning Checklist*

| OVERALL PLANNING ITEM | ANSWER |
|---|---|
| How many facilities (data centers) will you manage Opsware SAS? | |

*Table 1-1: Overall Planning Checklist*

| OVERALL PLANNING ITEM | ANSWER |
|---|---|
| In each of these facilities, how many servers will you manage with Opsware SAS? | |
| What is your naming convention for the Opsware facility names? (For example, you might use building or city names.) | |
| Have you taken an inventory of the operating systems and applications on the servers that you will manage with Opsware SAS? | |
| Which operating systems will you provision (install) with Opsware SAS? | |
| What applications will you provision (install) with Opsware SAS? | |
| Which Opsware SAS features will you use? | |
| What is your schedule for installing Opsware SAS core and for installing agents on the servers to be managed? | |
| Which of the following Opsware SAS architectures have you chosen?<br>• Standalone<br>• Multimaster mesh<br>• Satellite | |
| If you will be using multimaster mesh, how fast is the network connection between the Opsware cores? | |
| How many cores will you install? | |
| For each core, in which facility will it reside? | |
| How many Opsware Satellites will you install? | |

*Table 1-1: Overall Planning Checklist*

| OVERALL PLANNING ITEM | ANSWER |
|---|---|
| For each Satellite, in which remote facility will it reside? | |
| Which cores will the Satellite communicate with? | |
| How fast is the network connection between the Satellite and the core? | |
| Have you drawn a diagram showing the hosts that will run the Opsware core components? If applicable, the diagram should show the network connectivity between multimaster cores and between cores and Satellites. | |

### Specific Core Planning Checklist

The following checklist summarizes design decisions for a specific Opsware core installation.

*Table 1-2: Specific Core Planning Checklist*

| SPECIFIC CORE PLANNING ITEM | ANSWER |
|---|---|
| In which facility will this core reside? | |
| What will be the facility name? | |
| For the first core, what will be the facility ID and the default customer name? | |
| How many servers will this Opsware core manage? | |
| Will you distribute the Opsware core components across multiple servers? | |
| What are the host names of the servers on which the core components will be installed? | |

*Table 1-2: Specific Core Planning Checklist*

| SPECIFIC CORE PLANNING ITEM | ANSWER |
|---|---|
| For a multiple-server core, have you drawn a diagram that shows which components will run on which servers? | |
| For a multimaster mesh, will you be using an Opsware Software Repository Replicator? | |
| For a multiple-server core, will you have multiple instances of the following Opsware components?<br>• Data Access Engine<br>• Opsware Command Center (OCC)<br>• Media Server<br>• Global File System Server | |
| Will you deploy a load balancer on multiple instances of the following Opsware components?<br>• Data Access Engine<br>• Opsware Command Center (OCC) | |
| Will you install the following Opsware components into their own DMZ network?<br>• OS Provisioning Boot Server<br>• OS Provisioning Media Server | |
| Do you have the necessary licenses for Oracle? (The Opsware Model Repository uses an Oracle database.) | |
| Have you written your backup and recovery plan for the servers running Opsware SAS? | |

*Table 1-2: Specific Core Planning Checklist*

| SPECIFIC CORE PLANNING ITEM | ANSWER |
|---|---|
| Have you contacted your database administrator (DBA)? Your DBA will need to create an Oracle database before you install Opsware SAS. Also, your DBA will need to monitor the Oracle database when it goes into production. | |
| Have you contacted your network administrator? He or she will need to setup host name resolution (/etc/hosts, DNS) before the installation and will run a DHCP configuration tool after the installation. | |
| Which version of Opsware SAS are you installing? | |

### Specific Core Requirements Checklist

The following checklist summarizes the technical requirements that must be met before Opsware core installation.

*Table 1-3: Specific Core Requirements Checklist*

| REQUIREMENT | ANSWER |
|---|---|
| Have the hardware servers on which you will install the Opsware core components (core servers) been racked and stacked? | |
| Do you have root access to the core servers? | |
| Will you be able to mount Opsware SAS DVD and copy its contents to the core servers? | |
| Are the core servers running a supported operating system? | |
| Do the core servers meet the CPU requirements? | |

*Table 1-3: Specific Core Requirements Checklist*

| REQUIREMENT | ANSWER |
|---|---|
| Do the core servers meet the memory requirements? | |
| Do the core servers meet the disk space requirements? | |
| Are the servers for an individual core on the same LAN or VLAN? (Multimaster cores must be on separate VLANs.) | |
| Do the core servers have network connectivity to the servers they will manage? | |
| Have you verified that Network Information System (NIS) is *not* running on the core servers? | |
| If you will be using the Network File System (NFS) for Opsware components, such as the Software Repository or Media Server, does the root user have write access over NFS to the directories where the components are to be installed? | |
| Does the link speed and duplex of core and managed servers match the switch to which they are connected? | |
| Are the necessary TCP ports open on the core and managed servers? | |

## Pre-Installation Tasks Checklist

The following checklist summarizes the hands-on tasks you must perform before installing an Opsware core.

*Table 1-4: Pre-Installation Tasks Checklist*

| PRE-INSTALLATION TASK | TASK COMPLETED? |
|---|---|
| For the servers that will run the Opsware core components (core servers), perform the specific tasks for Linux and Solaris described in the section "Operating System Requirements" on page 29 (*Opsware® SAS Deployment and Installation Guide*). | |
| Set up the host name resolution (/etc/hosts or DNS) for the core servers. | |
| If network provisioning occurs on a separate network from the Opsware core components, you must set up DHCP proxying. | |
| Obtain `qchain.exe`, `mbsacli.exe`, and `mssecure.cab` from Microsoft and copy them to a location on your network that is accessible by the Opsware installer. | |
| Synchronize the system clocks on the core servers with an external Network Time Protocol (NTP) service. | |
| For a multimaster mesh, see the section "Prerequisites for a Multimaster Installation" on page 93 (*Opsware® SAS Deployment and Installation Guide*). | |

*Table 1-4: Pre-Installation Tasks Checklist*

| PRE-INSTALLATION TASK | TASK COMPLETED? |
|---|---|
| Create an Oracle database on the server on which you will install the Opsware Model Repository. See the Appendix, "Oracle Setup for Model Respository" (*Opsware® SAS Deployment and Installation Guide*). | |
| Verify that you have followed the instructions in Chapter 3, "Prerequisite Information for the Installer Interview" (*Opsware® SAS Deployment and Installation Guide*). | |

### Post-Installation Tasks Checklist

The following checklist summarizes the hands-on tasks you must perform after installing an Opsware core. For more information, see the "Post-Installation Tasks" chapter of the *Opsware® SAS Deployment and Installation Guide*.

*Table 1-5: Post-Installation Tasks Checklist*

| POST-INSTALLATION TASK | TASK COMPLETED? |
|---|---|
| Install the Windows Agent Deployment Helper. | |
| Configure DHCP for Opsware OS Provisioning. You may use the DHCP server included with Opsware SAS or an external DHCP server. | |
| For Windows OS provisioning, the host name `buildmgr` should resolve on Windows installation clients. | |
| For Patch Management on Windows NT or 2000, create a silent-installable version of IE 6.0 or later. | |

*Table 1-5: Post-Installation Tasks Checklist*

| POST-INSTALLATION TASK | TASK COMPLETED? |
|---|---|
| Multimaster mesh: Associate customers with the new facility. | |
| Multimaster mesh: Update the group permissions for the new facility. | |
| Multimaster mesh: Verify that the multimaster transaction traffic is flowing between the cores. | |

# Chapter 2: Pre-Installation Requirements

**IN THIS CHAPTER**

This section discusses the following topics:

- Operating System Requirements
- Network Requirements
- Patch Management Requirements
- Configuration Tracking Requirements
- Opsware Global File System (OGFS) Requirements
- Time and Locale Requirements

## Operating System Requirements

This section describes platform-specific requirements. For more information, see the "Hardware Requirements and Supported Operating Systems" section of the *Planning Deployments for Opsware® SAS*.

### Solaris Requirements

For Solaris, the Opsware core servers must meet the following requirements.

#### *Required Packages for Solaris*

The following listing of a Solaris Jumpstart profile shows the required packages:

```
cluster    SUNWCreq
cluster    SUNWCpm delete
package    SUNWadmap add
package    SUNWadmc add
package    SUNWesu add
package    SUNWswmt add
package    SUNWtoo add
package    SUNWtoox add
package    SUNWadmfw add
package    SUNWlibC add
package    SUNWlibCx add
```

```
package    SUNWinst add
package    SUNWucbt add
package    SUNWucbtx add
package    SUNWscpu add
package    SUNWscpux add
package    SUNWtcsh add
package    SUNWsacom add
package    SUNWpmr add
```

### *Other Solaris Requirements*

On the server where you will install the Opsware Command Center component, you must install the J2SE Cluster Patches for Solaris. You can download these patches from the following location:

```
http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/J2SE
```

On all core servers, verify that the Network File System (NFS) is configured and running.

## Linux Requirements

For Linux, the Opsware core servers must meet the following requirements:

### *Required Packages for Linux*

The following packages must be installed:

```
compat-db
compat-libstdc++
cpp
expat
gcc
glibc-devel
glibc-headers
glibc-kernheaders
kernel-source
libcap
libxml2-python
libstdc++-
libstdc++-devel
ncompress (contains uncompress utility)
nfs-utils
ntp
patch
patchutils
sharutils
strace
```

```
tcl
unzip
XFree86-libs
XFree86-libs-data
XFree86-Mesa-libGL
xinetd
zip
```

To verify that the `zip` package is installed, for example, enter the following command:

```
rpm -qa | grep zip
```

You can obtain the latest versions of these packages from the Red Hat errata web site.

### *Packages on Linux that Must Be Uninstalled*

If the Opsware core server already has the following applications installed, you must uninstall them before running the Opsware Installer.

```
samba
apache
rsync
httpd
tftp
dhcp
```

Existing versions of the `tftp` and `dhcp` packages cannot reside on the same server as the OS Provisioning Boot Server component, but they may reside on Opsware core servers that do not have the OS Provisioning Boot Server component.

To remove the `rsync` package, for example, enter the following command:

```
rpm -e --nodeps rsync
```

### *Other Linux Requirements*

For Linux systems, you must also perform the following tasks:

- Change the initial run level of the server to level 3 in the file `/etc/inittab`.
- If the server uses Integrated Drive Electronics (IDE) hard disks, enable Direct Memory Access (DMA) and some other advanced hard disk features to improve performance. Run the following script as root on the server, and then reboot the server:

  ```
  cat > /etc/sysconfig/harddisks << EOF
  USE_DMA=1
  MULTIPLE_IO=16
  EIDE_32BIT=3
  LOOKAHEAD=1
  EOF
  ```

## Network Requirements

This section discusses the following network requirements within a facility, open ports required for core components, and name resolution requirements. These requirements must be met for both standalone and multimaster cores.

### Network Requirements within a Facility

Before running the Opsware Installer, your environment must meet the following network requirements:

- The Opsware core servers must be on the same Local Area Network (LAN or VLAN).
- The Opsware core servers must have network connectivity to the servers that the Opsware core manages, and vice versa.
- The Opsware core servers cannot use the Network Information Service (NIS) for password and group databases. The Opsware components check for the existence of certain target accounts before creating them during installation.
- When using network storage for Opsware components, such as the Software Repository or Media Server, the network storage configuration must allow the root user to have write access over NFS to the directories where the components are to be installed.
- The speed and duplex mode of the NIC adapters of the Opsware core and managed servers must match the switch they are connected to. A mismatch causes poor network performance between the core and managed servers, making Opsware SAS unusable.

### Open Ports

Table 2-1 shows the ports that must be open on firewalls that protect the Opsware core components. The Gateway ports listed are the default values, which can be changed during the installation.

*Table 2-1: Open Ports on a Firewall Protecting an Opsware Core*

| PORT | COMPONENT | PURPOSE |
|---|---|---|
| 80 (TCP) | Opsware Command Center | HTTP redirector |
| 443 (TCP) | Opsware Command Center | OCC web, OCC Client, Opsware web services |
| 2001 (TCP) | Core Gateway | inbound tunnels from other Gateways |

*Table 2-1: Open Ports on a Firewall Protecting an Opsware Core (continued)*

| PORT | COMPONENT | PURPOSE |
|---|---|---|
| 2222 (TCP) | Opsware Global File System | global shell session from an SSH client |
| 3001 (TCP) | Agent Gateway | inbound Agent connections |
| 7580, 7581 (TCP) | Model Repository Multimaster Component | TIBCO Rendezvous web client |
| 8017 (UDP, TCP) | Agent Gateway | interface to the Build Manager |
| 8080 (TCP) | Opsware Command Center | OGFS Gateway for the OCC Client |

Table 2-2 shows the ports for the OS provisioning components that are accessed by servers during the provisioning process. (In Opsware SAS, provisioning refers to the installation of an operating system on a server.)

*Table 2-2: Open Ports for the OS Provisioning Components*

| PORT | COMPONENT | SERVICE |
|---|---|---|
| 67 (UDP) | Boot Server | DHCP |
| 69 (UDP) | Boot Server | TFTP |
| 111 (UDP, TCP) | Boot Server, Media Server | RPC (`portmapper`), required for NFS |
| Dynamic* | Boot Server, Media Server | `rpc.mountd`, required for NFS |
| 2049 (UDP, TCP) | Boot Server, Media Server | NFS |

* The `rpc.mountd` process runs on a dynamic port and is not fixed. Therefore, if a firewall is in place, it must be an application layer firewall that can understand the RPC request that the client uses to locate the port for `mountd`. The firewall must dynamically open that port.

Table 2-3 shows the ports that must be open on managed servers so that Opsware core servers can connect to managed servers.

*Table 2-3: Open Ports on Managed Servers*

| PORT | COMPONENT |
|---|---|
| 1002 (TCP) | Opsware Agent |

### Host and Service Name Resolution Requirements

Opsware SAS must be able to resolve Opsware server host names and service names to IP addresses through configuration of DNS or `/etc/hosts`.

#### *Previous Releases*

If you are installing Opsware components on servers where a previous release of Opsware SAS was installed (for example, 4.0), you must verify that the host names and service names resolve correctly as noted in this section.

#### *Opsware Core Servers and Name Resolution*

An Opsware core server must be able to resolve the fully qualified host name of itself and any other Opsware core server. (A fully qualified name includes the subdomain, for example, `myhost.acct.buzzcorp.com`.) Enter the `hostname` command and verify that it displays the fully qualified name.

Additionally, an Opsware core server must be able to resolve both the fully qualified and unqualified names of the Opsware services. (Each service name represents an Opsware component.) For example, both `truth` (unqualified) and `truth.acct.buzzcorp.com` (fully qualified) must resolve to the IP address of the server containing the Model Repository. The list of fully qualified names of the Opsware services follows:

- `truth.`*`subdomain`* - Model Repository
- `way.`*`subdomain`* - Command Engine
- `spin.`*`subdomain`* - Data Access Engine
- `theword.`*`subdomain`* - Software Repository
- `twist.`*`subdomain`* - Web Services Data Access Engine
- `occ.`*`subdomain`* - Opsware Command Center
- `buildmgr.`*`subdomain`* - OS Provisioning Build Manager

- `wordcache.`*`subdomain`* - Software Repository Multimaster Component (The name `wordcache` must resolve to the core server running the Software Repository.)

The Software Repository server must be able to resolve the IP address to the host name of the OGFS server. To enable this reverse lookup, configure DNS.

### DHCP Proxying

If network provisioning occurs on a separate network from the Opsware core components, you must set up DHCP proxying (for example, with Cisco IP Helper) to the DHCP server. If you set up DHCP proxying, the server/router performing the DHCP proxying must be the router for the network so that PXE will function correctly in the Opsware OS Provisioning Feature.

The Opsware Boot Server component includes a DHCP server, but does not include a DHCP proxy. You configure the DHCP server after installation by using the Opsware DHCP Network Configuration Tool. See *DHCP Configuration for OS Provisioning* in Chapter 5, on page 75.

### DMZ Network

The Boot Server and Media Server run various services (such as portmapper and rpc.mountd) that have been susceptible to network attacks. Opsware Inc. recommends that you segregrate the OS Provisioning Boot Server and Media Server components onto their own DMZ network. When you segregate these components, the ports listed previously) should be opened to the DMZ network from the installation client network. Additionally, the Boot Server and Media Server should have all vendor-recommended security patches applied.

## Patch Management Requirements

You must obtain several files from Microsoft and copy them to a directory that is accessible by the Opsware Installer. When you install the Opsware Software Repository, the Opsware Installer prompts you for the directory name.

Perform the following steps:

1. Obtain the following Microsoft Base Security Analyzer (MBSA) 1.2.1 files:

   - `qchain.exe`

     The `qchain.exe` utility is a command-line program that chains hotfixes together. Download the package containing `qchain.exe` from the following URL:

     ```
     http://www.microsoft.com/downloads/
     details.aspx?amp;displaylang=en&familyid=3C64D889-74F1-
     490B-A2FB-F15671A3B60C&displaylang=en
     ```

     Install the package on a Windows machine and locate the `qchain.exe` file.

   - `mssecure.cab`

     The `mssecure.cab` file contains the Microsoft patch database. Download `mssecure.cab` from the following URL:

     ```
     http://go.microsoft.com/fwlink/?LinkId=18922
     ```

   - `mbsacli.exe`

     Packaged with the MBSA 1.2.1 software, the `mbsaclie.exe` utility is a command-line program that performs security scans. Download MBSA 1.2.1 from the following URL:

```
http://download.microsoft.com/download/9/0/7/90769f0c-
c025-48bf-a9c7-60072d0cb717/MBSASetup-EN.msi
```

After the download, on a Windows machine run `MBSASetup-EN.msi` to install MBSA 1.2.1.

In the directory where you installed MBSA 1.2.1, locate the `mbascli.exe` file. By default, the file is installed here:

```
%program files%\Microsoft Baseline Security
Analyzer\mbascli.exe
```

2. Copy the MBSA 1.2.1 files you obtained in the preceding step to a directory that is accessible by the server where you will install the Opsware Software Repository. For example, you might copy the files to the following directory:

   `/home/win_util`

   The files that you want to copy are:

   ```
   qchain.exe
   mssecure.cab
   mbsacli.exe
   ```

3. Obtain the following Microsoft Base Security Analyzer (MBSA) 2.0 files:

   - `wsusscan.cab`

     The `wsusscan.cab` file contains the Microsoft patch database. Download `wsusscan.cab` from the following URL:

     ```
     http://go.microsoft.com/fwlink/?LinkId=39043
     ```

   - `WindowsUpdateAgent20-x86.exe`

     The `WindowsUpdateAgent20-x86.exe` file is required by the `mbsacli20.exe` utility. Download `WindowsUpdateAgent20-x86.exe` from the following URL:

     ```
     http://go.microsoft.com/fwlink/?LinkId=43264
     ```

   - `mbsacli20.exe`

     This utility is packaged with the MBSA 2.0 software as `mbsacli.exe`. In a later step, you will copy `mbsacli.exe` to `mbsacli20.exe`.

     The download files for MBSA 1.2.1 and MBSA 2.0 have the same name: `MBSASetup-EN.msi`. Before you download MBSA 2.0, rename the `MBSASetup-EN.msi` file you downloaded for MBSA 1.2.1.

     Download MBSA 2.0 from the following URL:

     ```
     http://www.microsoft.com/downloads/
     info.aspx?na=208&p=2&SrcDisplayLang=en&SrcCategoryId=&SrcF
     amilyId=4B4ABA06-B5F9-4DAD-BE9D-
     ```

```
7B51EC2E5AC9&u=http%3a%2f%2fdownload.microsoft.com%2fdownl
oad%2f3%2ff%2fd%2f3fd1a09d-af15-4ab7-a554-
0ac6c1e76c16%2fMBSASetup-EN.msi
```

After the download, on a Windows machine run `MBSASetup-EN.msi` to install MBSA 2.0. Do not overwrite the MBSA 1.2.1 installation. The default installation directories are different.

In the directory where you installed MBSA 2.0, locate the `mbsacli.exe` file. By default, the file is installed here:

```
%program files%\Microsoft Baseline Security Analyzer
2\mbsacli.exe
```

- `wusscan.dll`

  The `wusscan.dll` file is in the directory where you installed MBSA 2.0. By default, the file is here:

  ```
  %program files%\Microsoft Baseline Security Analyzer
  2\wusscan.dll
  ```

4. Copy the MBSA 2.0 `mbsacli.exe` file to `mbsacli20.exe` in the directory where you copied the files in step 2.

   Do *not* overwrite the MBSA 1.2.1 `mbsacli.exe` file you copied in step 2.

5. Copy the other three files you downloaded in step 3 to the directory where you copied the files in step 2. These other three files are:

   ```
   wsusscan.cab
   WindowsUpdateAgent20-x86.exe
   wusscan.dll
   ```

6. Verify that the destination directory contains the following files:

   ```
   mbsacli.exe
   mbsacli20.exe
   mssecure.cab
   qchain.exe
   WindowsUpdateAgent20-x86.exe
   wsusscan.cab
   wusscan.dll
   ```

7. Write down the name of the directory containing the files listed in the preceding step. When you install the Opsware Software Repository, you are prompted for the directory name. The Opsware Installer prompt is `windows_util_loc`.

During Opsware Agent installation, the files you obtained from Microsoft are downloaded from the Opsware Software Repository to the appropriate Windows servers. If newer versions of the files are uploaded to the Opsware Software Repository, they are downloaded to the managed servers during software registration.

## Configuration Tracking Requirements

When you run the Opsware Configuration Tracking feature in a facility, you must create a separate partition on the server running the Software Repository for the following Configuration Tracking directory:

```
/cust/word/<facility-name>/acsbar
```

The Configuration Tracking feature uses this directory to store the backup versions of tracked configuration files and databases.

## Opsware Global File System (OGFS) Requirements

This section discusses requirements of the OGFS.

### OGFS Store and Audit Hosts

When you run the Opsware Installer interviewer in advanced mode, you can specify values for the `ogfs.store.host` and `ogfs.audit.host` parameters. (See "Opsware Global File System Prompts" on page 59.) If you set either of these parameters to a host that runs neither the OGFS nor the Software repository, then perform the following steps on the host where you will install the OGFS:

1. With `mkdir`, create the directories that you specified for the `ogfs.store.path` and `ogfs.audit.path` parameters.
2. With a text editor, modify the `/etc/exports` file. For example:

   ```
   # Begin Opsware ogfs exports
      /cust/ogfs/store *(ro) 1.2.3.4(rw,no_root_squash)
      /cust/ogfs/audit *(ro) 1.2.3.4(rw,no_root_squash)
   # End Opsware ogfs exports
   ```

3. Run the following command:

   ```
   exportfs -a
   ```

### Name Service Caching Daemon (nscd) and OGFS

If the Name Service Caching Daemon (`nscd`) runs on the same server as the OGFS, then users cannot open a global shell session with a direct `ssh` connection. If `ncsd` is running on the OGFS server, the Opsware Installer turns it off and runs the `chkconfig nscd off` command to prevent it from starting after a reboot. No action by you is required.

## Time and Locale Requirements

This section discusses the time and locale requirements for core servers.

### Core Time Requirements

Opsware core servers (either standalone or multimaster) and Opsware Satellite servers must meet the following requirements. These time requirements do not apply to managed servers (that is, servers with Opsware Agents).

- Opsware core servers must maintain synchronized clocks. For example, you can synchronize the system clocks with an external server that uses NTP (Network Time Protocol) services.
- Opsware core servers must have their time zone set to Coordinated Universal Time (UTC).

On Linux servers, to configure the time zone, perform the following steps:

- Copy or link `/usr/share/zoneinfo/UTC` to `/etc/localtime`.
- Make sure that `/etc/sysconfig/clock` contains the following lines:

```
ZONE="UTC"
UTC=true
```

On Solaris servers, to configure the time zone, verify that `/etc/TIMEZONE` contains the following line:

```
TZ=UTC
```

### Locale Requirements

The core servers with the Model Repository and the Software Repository must have the `en_US.UTF-8` locale installed. To display data from managed servers in various locales, the core server with the Opsware Global File System (OGFS) must have those locales installed.

# Chapter 3: Prerequisite Information for the Installer Interview

**IN THIS CHAPTER**

This section discusses the following topics:

- Required Information for Running the Installer Interview
- Opsware Installer

## Required Information for Running the Installer Interview

The Opsware Installer interview prompts you for information about your environment that it saves in a response file. After the interview, the Opsware Installer reads the response file when it installs an Opsware core component onto a server.

Before you run the Installer interview, you must gather the information that you will enter for the interview prompts. Examples of this information are: the password for the Oracle `opsware_admin` user, the Opsware facility name for the core, and the Opsware authorization domain.

The Opsware Installer prompts you for a mode, either simple or advanced. In the simple mode, the Installer interview prompts you for fewer parameters.

The tables that follow list the various prompts that you will respond to when running the Installer interview. In the tables, prompts required only for the installation of a multimaster core are indicated by the word **Multimaster** (in bold font). Prompts required only for the advanced mode are denoted by the word **Advanced**.

### Model Repository Prompts

The Model Repository is the database that stores information about the hardware and software deployed in the operational environment. Most of the Model Repository prompts are for a standalone Opsware core. However, for multimaster mesh cores, you need to provide some additional information.

*Table 3-1: Model Repository Prompts*

| PROMPT | DESCRIPTION |
|---|---|
| Enter the service name (aka TNS name) of the Model Repository instance.<br><br>(Parameter: `truth.servicename`) | Specifies the service name, also known as the alias, for the Model Repository.<br><br>The service name can be determined by looking in the `tnsnames.ora` file on the Model Repository instance. The service name is the value before the first equals sign (=) in the file. The location of this file can vary, so check with your DBA if you are not sure where to look.<br><br>Source: The DBA who created the Oracle database.<br><br>Example: `truth.opsware.com` |
| Enter the service name (aka TNS name) of the Model Repository instance that you will be installing in the new facility.<br><br>(Parameter: `slaveTruth.servicename`) | **Multimaster**: Specifies the service name, also known as the alias, for the Model Repository of the target core.<br><br>The service name can be determined by looking in the `tnsnames.ora` file on the Model Repository instance. The location of this file can vary, so check with your DBA if you are not sure where to look.<br><br>Source: The DBA who created the Oracle database.<br><br>Example: `truth02.opsware.com` |

*Table 3-1: Model Repository Prompts (continued)*

| PROMPT | DESCRIPTION |
|---|---|
| Enter the SID of the Oracle instance that contains the Data Model Repository.<br><br>(Parameter: `truth.sid`) | **Multimaster**: Specifies the database system ID (SID) that was set when Oracle was installed on the server where the Model Repository is installed.<br><br>You can find out the SID by looking at the `tnsnames.ora` file. The location of this file can vary, so check with your DBA if you are not sure where to look.<br><br>Source: The DBA who created the Oracle database.<br><br>Example: `DTC05` |
| Enter the path of the Oracle home.<br><br>(Parameter: `truth.orahome`) | Specifies the base directory of the Oracle installation that was set when Oracle was installed.<br><br>You can determine the Oracle home directory by logging in as the `oracle user` on the Model Repository server, and checking the value of the `$ORACLE_HOME` environment variable. (For a remote database, this parameter refers to the installation of Oracle Client on the Model Repository server.)<br><br>Source: The DBA who created the Oracle database.<br><br>Example: `/cust/oracle/product/9.1` |
| Enter the path to the TNS admin directory (where the `tnsnames.ora` file resides).<br><br>(Parameter: `truth.tnsdir`) | Specifies the directory that contains the `tnsnames.ora` file. The location of the `tnsnames.ora` file can vary, so check with your DBA if you are not sure where to look.<br><br>Source: The DBA who created the Oracle database.<br><br>Example: `/var/opt/oracle` |

*Table 3-1: Model Repository Prompts (continued)*

| PROMPT | DESCRIPTION |
|---|---|
| Enter the full path to the directory where the export file will be saved.<br><br>(Parameter: `truth.dest`) | **Multimaster**: Specifies the directory where the database export file will be saved. This directory must exist on the Model Repository server in the source facility.<br><br>When adding a facility to a multimaster mesh, you must export the Model Repository from the source facility, then copy it to the destination facility.<br><br>Source: Arbitrary. (However, you must create the directory on the server before you run the Opsware Installer.)<br><br>Example: `/export/home/core1` |
| Enter the full path to the directory that contains the export file.<br><br>(Parameter: `truth.sourcePath`) | **Multimaster**: Specifies the directory on the Model Repository server in the destination facility where the export data file was copied from the source facility.<br><br>When adding a facility to a multimaster mesh, you must export the Model Repository data from the source facility, then copy it to the destination facility.<br><br>Source: Arbitrary. (However, the directory must exist on the server and contain the database export file before you run the Opsware Installer on that server.)<br><br>Example: `/export/home/core2` |
| Please enter the IP address of the device where you are planning to install the Model Repository in the new facility.<br><br>(Parameter: `slaveTruth.truthIP`) | **Multimaster**: Specifies the IP address of the host on which you will install the Model Repository for the new target core.<br><br>Source: Arbitrary.<br><br>Example: `192.168.165.242` |

*Table 3-1: Model Repository Prompts (continued)*

| PROMPT | DESCRIPTION |
|---|---|
| Please enter the IP address of the device where you are planning to install the Multimaster Infrastructure Components (vault).<br><br>(Parameter: `slaveTruth.vaultIP`) | **Multimaster**: Specifies the IP address of the host on which you will install the Multimaster Infrastructure Components for the core.<br><br>Source: Arbitrary.<br><br>Example: `192.168.165.242` |

### Database (Model Repository) Password Prompts

To ensure a secure installation of Opsware SAS, the Opsware Installer prompts you to set passwords for numerous Oracle user accounts that the Opsware components use to interact with one another. The passwords must meet standard Oracle criteria, as follows:

- The password cannot contain an Oracle reserved word (see Oracle's documentation for a full list).
- The password must be between 1 and 30 characters long.
- The password must start with a letter and use only alphanumeric and underscore (_) characters.

*Table 3-2: Database Password Prompts*

| PROMPT | DESCRIPTION |
|---|---|
| Enter database password for the `opsware_admin` user.<br><br>(Parameter: `truth.oaPwd`) | Specifies the `opsware_admin` password created by your database administrator.<br><br>`opsware_admin` is an Oracle user that the Opsware Installer uses during installation to perform certain functions.<br><br>Source: This must be the password that your DBA set for the `opsware_admin` user when setting up the Oracle instance on the server where you will install the Model Repository. |

*Table 3-2: Database Password Prompts (continued)*

| PROMPT | DESCRIPTION |
|---|---|
| Enter database password for the lcrep user.<br><br>(Parameter: `truth.lcrepPwd`) | **Advanced**: Sets the password for the `lcrep` database user.<br><br>The Opsware Installer automatically creates an Oracle user `lcrep`, which Opsware SAS uses internally for running multimaster replication between Opsware cores.<br><br>Source: Arbitrary. (However, must meet the requirements for Oracle passwords.)<br><br>Example: `x145_pwd03` |
| Enter database password for the gcadmin user.<br><br>(Parameter: `truth.gcPwd`) | Sets the password for the `gcadmin` database user.<br><br>The Opsware Installer automatically creates an Oracle user `gcadmin`, which Opsware SAS uses internally for removing old data from certain tables (referred to as the garbage collection process).<br><br>Source: Arbitrary. (However, must meet the requirements for Oracle passwords.)<br><br>Example: `x145_pwd03` |
| Enter the database password for the truth user.<br><br>(Parameter: `truth.truthPwd`) | **Advanced**: Sets the password for the he `truth` user.<br><br>The Opsware Installer automatically creates this Oracle user, which is the main schema owner for the Model Repository.<br><br>Source: Arbitrary. (However, must meet the requirements for Oracle passwords.)<br><br>Example: `x145_pwd03` |

*Table 3-2: Database Password Prompts (continued)*

| PROMPT | DESCRIPTION |
|---|---|
| Enter the database password for the spin user.<br><br>(Parameter: `truth.spinPwd`) | **Advanced**: Sets the password for the `spin` user.<br><br>The Opsware Installer automatically creates this database user.<br><br>Source: Arbitrary. (However, it must meet the requirements for Oracle passwords.)<br><br>Example: `x145_pwd03`<br><br>Note: Passwords for the `spin` user must be the same across all the cores in the mesh. |
| Enter the database password for the twist user.<br><br>(Parameter: `truth.twistPwd`) | **Advanced**: Sets the password for the `twist` user.<br><br>The Opsware Installer automatically creates this user.<br><br>Source: Arbitrary. (However, it must meet the requirements for Oracle passwords.)<br><br>Example: `x145_pwd03` |
| Enter the database password for the vault user.<br><br>(Parameter: `truth.vaultPwd`) | **Multimaster**: Sets the Model Repository, Multimaster Component password. This prompt only appears when installing Opsware SAS in multimaster mode.<br><br>The Opsware Installer automatically creates the `vault` user.<br><br>The Model Repository, Multimaster Component propagates and synchronizes changes from each Model Repository database to all other Model Repository databases.<br><br>Source: Arbitrary. (However, it must meet the requirements for Oracle passwords.)<br><br>Example: `x145_pwd03` |

*Table 3-2: Database Password Prompts (continued)*

| PROMPT | DESCRIPTION |
|---|---|
| Enter the database password for the public views user.<br><br>(Parameter: `truth.pubViewsPwd`) | **Advanced**: Sets the password for the `public_views` user, which Opsware SAS uses for the Data Center Intelligence (DCI) module (server reporting). The DCI module uses this password when connecting with the Model Repository. The Opsware Installer automatically creates the public views user.<br><br>If you are using Brio, Crystal Reports, or other data reporting tools with the DCI module, you are asked for the database user password when you log into those applications so that you have read-only access to the Model Repository data.<br><br>Source: Arbitrary. (However, it must meet the requirements for Oracle passwords.)<br><br>Example: `x145_pwd03` |
| Enter the database password for the AAA user.<br><br>(Parameter: `truth.aaaPwd`) | **Advanced**: Sets the password for the AAA user, which Opsware SAS uses for the Access, Authentication, and Authorization (AAA) feature. The Opsware Installer automatically creates the AAA user.<br><br>Source: Arbitrary. (However, it must meet the requirements for Oracle passwords.)<br><br>Example: `x145_pwd03` |
| Please enter the password to use for DCML exchange tool user.<br><br>(Parameter: `truth.detuserpwd`) | **Advanced**: Sets the password for the `DETUSER`, which Opsware SAS uses for the DCML Exchange Tool (DET). The Opsware Installer automatically creates the `DETUSER`.<br><br>Source: Arbitrary. (However, it must meet the requirements for Oracle passwords.)<br><br>Example: `x145_pwd03` |

### Opsware Component Password Prompts

This section lists the password prompts for the components other than the Model Repository.

In a multimaster mesh, the following passwords set during the Opsware Installer interview must be the same in all cores belonging to the mesh.

*Table 3-3: Component User and Password Prompts*

| PROMPT | DESCRIPTION |
|---|---|
| Enter the password for Build Manager user.<br><br>(Parameter: `twist.buildmgr.passwd`) | **Advanced**: Sets the password for the `buildmgr` user that the `buildmgr` process will use when connecting to and authenticating with the Web Services Data Access Engine. The Opsware Installer automatically creates this user.<br><br>The password cannot contain spaces or a forward slash (/).<br><br>Source: Arbitrary.<br><br>Example: `x145_pwd03` |
| Enter the password for Integration user.<br><br>(Parameter: `twist.integration.passwd`) | **Advanced**: Sets the password for the `integration` user that a customer can use to access the SOAP APIs on the Web Services Data Access Engine. The Opsware Installer automatically creates the `integration` user.<br><br>The password cannot contain a forward slash (/).<br><br>Source: Arbitrary.<br><br>Example: `x145_pwd03` |

*Table 3-3: Component User and Password Prompts (continued)*

| PROMPT | DESCRIPTION |
|---|---|
| Enter the password to decrypt cryptographic material.<br><br>(Parameter: `decrypt_passwd`) | Sets the password to use for decrypting cryptographic material. It cannot contain any spaces. The password must be between 4 and 20 characters long.<br><br>This password must be the same across all Opsware cores in a multimaster mesh.<br><br>Source: Arbitrary.<br><br>Example: `x145_pwd03` |
| Enter the password to use for admin entry.<br><br>(Parameter: `cast.admin_pwd`) | Sets the password for the Opsware `admin` user. The password cannot contain any spaces. The Opsware Installer automatically creates the `admin` user.<br><br>When you log into the Opsware Command Center in the facility, you log in as the `admin` user supply the password you provide at this prompt.<br><br>In general, you will *not* need to log into the directory manager (Netscape Directory Server) by using this user and password unless you need to troubleshoot directory issues.<br><br>Source: Arbitrary.<br><br>Example: `x145_pwd03` |

### Facility Prompts

A facility refers to the collection of servers that a single Opsware core manages. If you are performing a standalone core installation, your deployment is made up of a single facility. Multimaster installations, however, make up two or more facilities: one facility for each core that you install.

*Table 3-4: Facility Prompts*

| PROMPT | DESCRIPTION |
|---|---|
| Enter the authorization domain (uppercase).<br><br>(Parameter: `truth.authDom`) | Sets the authorization domain for the initial (default) customer. This value is usually the same as the domain name. It must be uppercase, less than 50 characters, and in domain name format.<br><br>You must use the same value for every Opsware core in your multimaster mesh. The Opsware Installer only prompts you for this value when you are installing your first, standalone Opsware core.<br><br>Source: Arbitrary.<br><br>Example: `XYZ.COM` |
| Enter the subdomain for this facility (lowercase, no spaces).<br><br>(Parameter: `truth.dcSubDom`) | Specifies the fully-qualified DNS subdomain where the Opsware core is deployed.<br><br>This value must be unique for each core in the multimaster mesh. The value is based on the VLAN for the facility in which you are installing the Opsware core.<br><br>It must be lowercase, less than 50 characters, and in subdomain format.<br><br>Source: Your network administrator.<br><br>Example: `dc1.opsware.com` |

*Table 3-4: Facility Prompts (continued)*

| PROMPT | DESCRIPTION |
|---|---|
| Please enter the subdomain for the facility you are about to create (lowercase, no spaces).<br><br>(Parameter: `slaveTruth.dcSubDom`) | **Multimaster**: Specifies the fully-qualified DNS subdomain where the target core is deployed.<br><br>This value must be unique for each core in the multimaster mesh. The value is based on the VLAN for the facility in which you are installing the target core.<br><br>It must be lowercase, less than 50 characters, and in subdomain format.<br><br>Source: Your network administrator.<br><br>Example: `dc2.opsware.com` |
| Enter the facility short name (uppercase, no spaces).<br><br>(Parameter: `truth.dcNm`) | Sets the default facility in the core.<br><br>Some Opsware SAS processes use this name internally. It must be uppercase, less than 25 characters, and cannot contain spaces or special characters (although dashes and underscores are allowed).<br><br>Source: Arbitrary.<br><br>Example: HEADQUARTERS |
| Please enter the short name of the new facility you would like to define<br><br>(Parameter: `slaveTruth.dcNm`) | Sets the default facility in the target core.<br><br>Some Opsware SAS processes use this name internally. It must be less than 25 characters, and cannot contain spaces or special characters (although dashes and underscores are allowed).<br><br>Source: Arbitrary.<br><br>Example: NORTHSIDE |

*Table 3-4: Facility Prompts (continued)*

| PROMPT | DESCRIPTION |
|---|---|
| Enter the default locale for users of the Opsware Command Center.<br><br>(Parameter: `default_locale`) | Specifies the default locale (language, character sets, and date and time formats) for the Opsware SAS core.<br><br>Source: In this release, the allowed values are `en` (English) and `ja` (Japanese).<br><br>Example: `en` |
| Enter the facility long name.<br><br>(Parameter: `truth.dcDispNm`) | **Advanced**: Sets the name that displays in the Opsware Command Center.<br><br>It must be unique, less than 50 characters, and cannot include any special characters (< > & * \ ' ?).<br><br>Source: Arbitrary.<br><br>Example: Los Angeles Office |
| Please enter the long name for the facility that you are adding to the mesh.<br><br>(Parameter: `slaveTruth.dcDispNm`) | **Multimaster, Advanced**: Sets the name of the target core that displays in the Opsware Command Center.<br><br>It must be unique, less than 50 characters, and cannot include any special characters (< > & * \ ' ?).<br><br>Source: Arbitrary.<br><br>Example: Toronto Office |

*Table 3-4: Facility Prompts (continued)*

| PROMPT | DESCRIPTION |
|---|---|
| Enter the facility ID (number only, less than 1000, with no leading zeros).<br><br>(Parameter: `truth.dcId`) | Specifies the ID that uniquely identifies a facility.<br><br>When you install a standalone core, you choose the facility ID during the installer interview.<br><br>When you install a target core in a multimaster mesh, the facility ID is automatically generated when you add the facility in the Opsware Command Center. You specify this automatically-generated ID during the installer interview.<br><br>Find the target facility ID by logging into the Opsware Command Center at the source facility. Select Opsware Facilities under Environment in the navigation panel and click the facilities' name.<br><br>**REQUIREMENT**<br>Opsware facility IDs must be less than 1000. Therefore, you must specify a number for the first facility that is well below 1000 so you can continue to add facilities to your multimaster mesh. If the Opsware Command Center automatically generates a number that is 1000 or higher, the installation will fail.<br><br>Source: Arbitrary for the first facility; set by the Opsware SAS for subsequent facilities.<br><br>Example: 100 |
| Enter the customer name (uppercase, no spaces).<br><br>(Parameter: `truth.acctNm`) | Sets the default customer for the facility in which you are installing the core.<br><br>Some Opsware SAS processes use this name internally. It cannot be OPSWARE or CUSTOMER. It must be a unique name, less than 35 characters, and cannot contain spaces.<br><br>Source: Arbitrary.<br><br>Example: MARKETING |

*Table 3-4: Facility Prompts (continued)*

| PROMPT | DESCRIPTION |
|---|---|
| Enter the customer display name.<br><br>(Parameter: `truth.acctDispNm`) | **Advanced**: Sets the display name in the Opsware Command Center for the initial (default) customer.<br><br>The name must be unique, less than 50 characters, and cannot include any special characters (< > & * \ ' ?).<br><br>Source: Arbitrary.<br><br>Example: IT Department |

## OS Provisioning and Patch Management Prompts

The response to the prompt for the windows utilities directory depends on the steps you performed in "Patch Management Requirements" on page 36.

*Table 3-5: OS Provisioning and Patch Management Prompts*

| PROMPT | DESCRIPTION |
|---|---|
| Please enter the directory that contains Microsoft's qchain.exe, mbsacli.exe, mssecure.cab, wusscan.dll, mbsacli20.exe, WindowsUpdateAgent20-x86.exe and wsusscan.cab files<br><br>(Parameter: `windows_util_loc`) | Specifies the directory to which you've copied the Microsoft utilities required for the Patch Management feature on Windows.<br><br>Source: Arbitrary. (However, this directory must exist on the server where the Software Repository is installed.)<br><br>Example: `/home/win_util` |
| Enter the OS Provisioning Boot Server IP or host name.<br><br>(Parameter: `bootagent.host`) | Specifies the server on which you will install the OS Provisioning Boot Server component.<br><br>You must provide a valid IP address or host name that can be resolved from the server on which you installed the OS Provisioning Boot Server and the Build Manager. Additionally, the host name must be resolvable by Opsware managed servers for OS provisioning. |

*Table 3-5: OS Provisioning and Patch Management Prompts (continued)*

| PROMPT | DESCRIPTION |
|---|---|
| Enter the host name or IP of the Build Manager.<br><br>(Parameter: `boot_server.buildmgr_host`) | Specifies the server on which you will install the OS Provisioning Build Manager.<br><br>You must provide a valid IP address or host name that can be resolved from the server on which you install the OS Provisioning Boot Server. |
| Enter the default network speed/duplex setting for Solaris servers.<br><br>(Parameter: `boot_server.speed_duplex`) | Sets the default network speed and duplex that will be used by Solaris servers booted from this boot server during Opsware OS provisioning. Valid responses are 100fdx, 100hdx, 10fdx, 10hdx, 100T4, and autoneg.<br><br>Enter a value without spaces.<br><br>Source: Arbitrary.<br><br>Example: `100fdx` |
| Enter the pathname of the Red Hat Linux media.<br><br>(Parameter: `media_server.linux_media`) | Specifies the path to the Linux OS media on the server on which the Software Repository will be installed.<br><br>Providing the path to the Linux OS media does not actually copy the media to this host.<br><br>See the *Opsware® SAS User's Guide* for the steps required to set up the media on the Media Server.<br><br>Source: Arbitrary. (However, this directory must exist on the server where the Software Repository is installed.)<br><br>Example: `/home/os_media/linux/` |

*Table 3-5: OS Provisioning and Patch Management Prompts (continued)*

| PROMPT | DESCRIPTION |
|---|---|
| Enter the pathname of the Solaris media.<br><br>(Parameter: `media_server.sunos_media`) | Specifies the path to the Sun Solaris OS media on the server on which the Software Repository will be installed.<br><br>Providing the path to the Solaris OS media does not actually copy the media to this host.<br><br>See the *Opsware® SAS User's Guide* for the steps required to set up the media on the Media Server.<br><br>Source: Arbitrary. (However, this directory must exist on the server where the Software Repository is installed.)<br><br>Example: `/home/os_media/solaris/` |
| Enter the pathname of the Windows media.<br><br>(Parameter: `media_server.windows_media`) | Specifies the path to the Microsoft Windows OS media on the server on which the Software Repository will be installed.<br><br>The OS Provisioning feature exports Windows OS media to SMB clients through a Samba share.<br><br>Providing the path to the Windows OS media does not actually copy the media to this host.<br><br>See the *Opsware® SAS User's Guide* for the steps required to set up the media on the Media Server.<br><br>Source: Arbitrary. (However, this directory must exist on the server where the Software Repository is installed.)<br><br>Example: `/home/os_media/windows/` |
| Enter the share name to use for the Windows media sharing server.<br><br>(Parameter: `media_server.windows_share_name`) | **Advanced**: Sets the share name that you want Samba to use to export the Windows OS media.<br><br>The share name is not case sensitive.<br><br>Source: Arbitrary.<br><br>Example: `WINMEDIA` |

*Table 3-5: OS Provisioning and Patch Management Prompts (continued)*

| PROMPT | DESCRIPTION |
|---|---|
| Enter a password to write-protect the Windows media share. Import_media prompts for this password each time it is run.<br>(Parameter: `media_server.windows_share_password`) | **Advanced**: Sets the root user password, which enables write access to the Windows share. The Opsware Import Media Tool prompts for this password each time it is run.<br>The password cannot contain spaces.<br>Source: Arbitrary.<br>Example: `x145_pwd03` |

### Opsware Gateway Prompts

These prompts are for the IP addresses and ports at which Opsware Gateways can be contacted by core components, Agents, or other Opsware Gateways. The port number must be less that 64001.

*Table 3-6: Opsware Gateway Prompts*

| PROMPT | DESCRIPTION |
|---|---|
| Please enter the port on which the administrative interface for the core gateway will run.<br>(Parameter: `cgw_admin_port`) | **Advanced**: Specifies the port of the Opsware Gateway's administrative interface, which allows you to view the configuration and monitor traffic flow.<br>Source: Arbitrary.<br>Example: 8085 |
| Please enter the IP address of the core Opsware Gateway.<br>(Parameter: `cgw_address`) | Specifies the IP address of the Opsware Gateway in the core at which other core components and Gateways can contact the core. In an Opsware Satellite installation, this IP address points to the core Gateway contacted by the Satellite.<br>Source: Arbitrary.<br>Example: `192.168.165.242` |

*Table 3-6: Opsware Gateway Prompts (continued)*

| PROMPT | DESCRIPTION |
|---|---|
| Please enter the port on which core components can contact this gateway to request tunneled connections.<br>(Parameter: `cgw_proxy_port`) | **Advanced**: Specifies the port of the Opsware Gateway in the core at which components in the same core can request connections to other components.<br>Source: Arbitrary.<br>Example: 3002 |
| Please enter the port on which Agents can contact the gateway to request connection to core components.<br>(Parameter: `agw_proxy_port`) | Specifies the port of the Opsware Gateway in the core at which Opsware Agents can request connections to core components.<br>Source: Arbitrary.<br>Example: 3001 |
| Please enter the port on which this gateway will listen for connections from other gateways.<br>(Parameter: `cgw_tunnel_listener_port`) | Specifies the port at which this Opsware Gateway will listen for connections from other Opsware Gateways.<br>Source: Arbitrary.<br>Example: 2001 |

### Opsware Global File System Prompts

The following prompts are for specifying IP addresses and directories for the Opsware Global File System.

*Table 3-7: Opsware Global File System Prompts*

| PROMPT | DESCRIPTION |
|---|---|
| Please enter the IP or host name of the nfs server for the Opsware Global File System user home and tmp directories.<br>(Parameter: `ogfs.store.host`) | **Advanced**: Specifies the server from which the storage for the home and tmp directories for the Opsware Global File System will be mounted.<br>Source: Arbitrary.<br>Example: `192.168.198.92` |

*Table 3-7: Opsware Global File System Prompts (continued)*

| PROMPT | DESCRIPTION |
|---|---|
| Please enter the absolute path on the nfs server for the Opsware Global File System user home and tmp directories.<br>(Parameter: `ogfs.store.path`) | **Advanced**: Specifies the directory for the storage of the home and tmp directories of the Opsware Global File System.<br>Source: Arbitrary.<br>Example: `/cust/ogfs/store` |
| Please enter the IP or host name of the nfs server for the Opsware Global File System where the audit streams will be stored.<br>(Parameter: `ogfs.audit.host`) | **Advanced**: Specifies the IP address of the server where storage for audit streams for the Opsware Global File System will be mounted.<br>Source: Arbitrary.<br>Example: `192.168.165.242` |
| Please enter the absolute path on the nfs server for the Opsware Global File System where the audit streams will be stored.<br>(Parameter: `ogfs.audit.path`) | **Advanced**: Specifies the path for the storage of the audit streams for the Opsware Global File System.<br>Source: Arbitrary.<br>Example: `/cust/ogfs/audit` |
| Please enter comma-separated list of IP address(es) for the devices where the Opsware Global File System (OGFS) is going to be installed in this facility (ip,ip...).<br>(Parameter: `hub.ip`) | Specifies one or more IP addresses of the servers on which to install the Opsware Global File System.<br>Multiple entries are separated by commas.<br>Source: Arbitrary.<br>Example: `192.168.198.92` |

### Uninstallation Prompts

The prompts in the following table appear when you are uninstalling an Opsware core.

*Table 3-8: Uninstallation Prompts*

| PROMPT | DESCRIPTION |
|---|---|
| Do you need to preserve any of the data in this database?<br>(Parameter: `truth.uninstall.needdata`) | Because uninstalling the Model Repository permanently deletes all data in the database, the uninstallation process stops if you answer yes to this parameter, so you have the opportunity to back up the data you would like to preserve. The Opsware Installer does not preserve any data.<br>Example: `y` |
| Are you sure you want to remove all data and schema from this database?<br>(Parameter: `truth.uninstall.aresure`) | Because uninstalling the Model Repository permanently deletes all data in the database, the uninstallation process stops if you answer no to this parameter. |
| Would you like to preserve the database of cryptographic material?<br>(Parameter: `save_crypto`) | If you answer yes, the database of cryptographic material is saved. Otherwise, it is deleted when the uninstallation finishes.<br>Example: `y` |
| Are you absolutely sure you want to remove all packages in the repository?<br>(Parameter: `word.remove_files`) | If you answer yes, the packages, logs, and cryptographic material for the Software Repository are removed.<br>Example: `y` |

## Opsware Installer

This section discusses the following topics:

- Installation Media for the Opsware Installer
- Opsware Installer Command Line Syntax
- Installer Interview

- Opsware Installer Logs

### Installation Media for the Opsware Installer

The Opsware SAS is available on and installable from a DVD, which contains the scripts for installing, uninstalling, and upgrading components. For the script names, see "Opsware Installer Command Line Syntax" on page 62. The remaining DVDs contain the packages used in the installation of the Opsware components. If you are installing the Opsware SAS directly from a DVD, the Opsware Installer prompts you to change the DVD that is loaded in the server you are installing a component on.

#### *Copying the DVD to a Local Disk*

Opsware Inc. recommends that you copy the contents of the Opsware SAS DVD to a local disk or to a network share and run the Opsware Install from that location. When you copy the contents of the DVD to a local disk or the network, you must create a directory structure that duplicates the structure of the DVD, for example:

```
/opsware_system
```

The path of the directory where you copy the contents of the DVD cannot have spaces.

When you run the Opsware Installer from the common parent directory, `/opsware_system`, the Opsware Installer switches automatically to the directory it needs to complete the part of the installation process that it is currently performing.

### Opsware Installer Command Line Syntax

The Opsware Installer is run by using one of the following three scripts:

- `install_opsware.sh` - installs a component
- `upgrade_opsware.sh` - upgrades a component
- `uninstall_opsware.sh` - uninstalls a component

All three of these scripts run with the same command line options, as the following table shows.

*Table 3-9: Opsware Installer Command Line Options*

| OPTION | DESCRIPTION |
|---|---|
| `-h` | Display the Opsware Installer help for the command line options.<br>To display help during the interview, press `ctrl-I`. |
| `--resp_file=`*`file`*<br>`(-r` *`file`*`)` | Install an Opsware component, using the values in the specified response file.<br>The installer prompts for the component to install and then runs an interview that only prompts for data missing in the response file. If the response file is incomplete, the installer prompts for the missing information.<br>The installer keeps an inventory of the components that are installed on a given server. |
| `--interview` | Conduct the installation interview to obtain values for component parameters. At the end of the interview, the installer saves the values in the response file.<br>Usually, you specify this option when you run the Opsware Installer on the host where the Model Repository has been or will be installed. You also specify this option when you have a complete response file but need to run the installer in a different mode, such as converting a standalone core to multimaster.<br>If you specify both the `--interview` and `--resp_file` options, the installer runs the interview, using the values in the response file as the defaults.<br>If you specify no command line options, the installer runs as if you specified the `--interview` option. |
| `--verbose` | Run the installer in verbose mode. |

### Installer Interview

The interview prompts you for the mode, either simple or advanced. In the simple mode, the interview does not prompt for parameters that are rarely modified. (Such parameters include the various Oracle passwords used internally by the Opsware components.) If you use the simple mode, the installer will use default values for these parameters. In the advanced mode, the installer prompts for all parameters that are relevant to the type of installation.

The installer validates responses to the interview prompts as you enter them; you are asked to re-enter a value until the installer is able to validate the answer. Some parameters are also revalidated during the actual installation of components. If a response to a prompt cannot be validated at installation, the installer runs a mini-interview.

At any time during the interview, you can press `ctrl-I` to display help for the current prompt.

After all parameters have values, the installer asks if you want to finish the interview. If you want to go back and review or change your answers, press n. If you press y, the installer prompts for the name of the response file in which it will save your answers. (The directory containing the response file must exist.) After saving the file, the installer asks if you'd like to continue the installation using the data from the response file. If you press y, the installer displays the Opsware components to install. If you press n, the installer exits.

When you install a core on multiple servers, you should copy the response file to the other servers so that the installations of subsequent components can use the data in the response file.

### Opsware Installer Logs

Each time you run the Opsware Installer, it generates the following log file:

`/var/lc/install_opsware/install_opsware.`*`timestamp`*`.log`

If you specify the --verbose option, the following log file is created:

`/var/lc/install_opsware/install_opsware.`*`timestamp`*`_verbose.log`

Some components have supplementary logs that contain additional details about the installation of those components.

The installation of the Model Repository creates the following log files:

`/var/lc/install_opsware/truth/truth_install_`*`number`*`.log`
`/var/lc/install_opsware/truth/truth_install_`*`number`*`_verbose.log`

# Chapter 4: Opsware Standalone Installation

**IN THIS CHAPTER**

This section discusses the following topics:

- Overview of the Standalone Installation
- Prerequisites for Installing a Standalone Core
- Installing a Standalone Core
- Opsware Command Center Web Client
- Logging into the Opsware Command Center

## Overview of the Standalone Installation

A standalone core manages servers in a single facility. The following steps provide an overview of the standalone installation process. For detailed instructions, see "Prerequisites for Installing a Standalone Core" on page 66.

1. Create an Oracle database on the server where you will install the Opsware Model Repository. (For a remote database, install Oracle Client on the Model Repository server.) See "Oracle Setup for Model Repository" on page 141.
2. Obtain the Opsware SAS installation DVD.
3. Run the Opsware Installer (`install_opsware.sh` script) in interview mode. The interviewer prompts you for information about your environment and saves the information in a response file.
4. Run the Opsware Installer and select the Opsware components to install. In this step, the Installer creates the Opsware directories and files on a server. For a single-server installation, you only need to run the Installer once. For a multiple servers, you log on to each server and run the Installer, specifying the components to install. You must install the Opsware core components in the order displayed by the Opsware Installer (see step 13 on page 68).

## Prerequisites for Installing a Standalone Core

Before you install a standalone core, you must perform the following tasks:

- Plan your Opsware System deployment. When planning for a core, you must decide whether you want to install the core components on a single server or on multiple servers. See the *Planning Deployments for Opsware® SAS*.
- Perform the pre-installation administration tasks such as configuring the network. See "Pre-Installation Requirements" on page 29.
- Create an Oracle database, which is required for the Opsware Model Repository. See "Oracle Setup for Model Repository" on page 141.
- Gather information in preparation for the Opsware Installer interview. This information includes the name and ID of the facility for the core. See "Prerequisite Information for the Installer Interview" on page 41.

## Installing a Standalone Core

This section contains step-by-step instructions for running the Opsware Installer (`install_opsware.sh` script).

1. Obtain the Opsware Server Automation System (SAS) installation media.

   See "Installation Media for the Opsware Installer" on page 62, including the recommendation, "Copying the DVD to a Local Disk."

2. On each server where you will install the new Opsware core, mount the DVD or NFS-mount the directory that contains a copy of the DVD contents.

   The Opsware Installer must have read/write root access to the directories where it installs Opsware components, even NFS-mounted network appliances.

3. On the server where you want to install the Opsware Model Repository, in a terminal window log in as root.

Before installing the Model Repository, you must create and start up an Oracle database on the server that will run the Model Repository. (For a remote database, install Oracle Client on the Model Repository server.) See "Oracle Setup for Model Repository" on page 141.

4 Change to the root directory:

```
cd /
```

5 Run the Opsware Installer in interview mode by invoking it with no command-line options:

```
/opsware_system/opsware_installer/install_opsware.sh
```

You must specify the full path to the script. The directory path shown in this step indicates that you copied the Opsware SAS DVDs to a local disk or network share using the required directory structure.

The Opsware Installer the displays following options:

```
Welcome to the Opsware Installer. Please select one of the
following installation options:

1 - Standalone Installation: Standalone Opsware Core
2 - Multimaster Installation: First Core (convert from
standalone)
3 - Multimaster Installation: Define New Facility; Export
Model Repository
4 - Multimaster Installation: Additional Core
5 - Satellite Installation: Opsware Satellite
```

6 At the installation options prompt, select the following option:

```
1 - Standalone Installation: Standalone Opsware Core
```

7 At the interview mode prompt, select one of the following options:

```
1 - Simple Interview Mode
2 - Advanced Interview Mode
```

Option 1 is for using default values for many of the configuration parameters. Option 2 is for specifying all configuration parameters during the interview.

8 Respond to the interview prompts.

The installer displays default values in square brackets [].

See “Required Information for Running the Installer Interview” on page 41.

When you run the interview, the paths for the OS provisioning media must already exist on the server where you will install the OS Provisioning Media Server component.

9 Decide if you want to finish the interview.

When you enter all of the required information, the Opsware Installer displays this message:

```
All parameters have values. Do you wish to finish the
interview (y/n):
```

If you are satisfied with your answers, press y.

If you want to review or change your answers, press n. The installer displays the prompts again, showing in brackets [ ] the values that you previously entered.

If you are satisfied with your answers, press y.

**10** Create the response file.

When you are finished with the interview, the installer prompts you for the name of the response file:

```
Name of response file to write
[/usr/tmp/oiresponse.stand_single]
```

The response file is a text file that contains the answers you entered during the interview. You can enter the full path and name of the response file or accept the default. In either case, write down the name of the response file. Note that the default file name corresponds to the type of installation.

**11** The Opsware Installer prompts you to indicate whether you want to continue the installation by using the response file:

```
Would you like to continue the installation using this
response file? (y/n):
```

If you are satisfied with the responses you entered in the interview and you are ready to install the Model Repository now, enter `y` to continue. If you do not want to install the Model Repository now, enter `n`.

**12** If you entered `y` in the previous step, skip this step. If you entered `n` in the previous step, invoke the Opsware Installer with the `-r` option to specify the response file created by the interview:

```
/opsware_system/opsware_installer/install_opsware.sh -r
<full_path_to_response_file>
```

**13** At the components prompt, select one or more components to install:

```
Welcome to the Opsware Installer.
Please select the components to install.
1 ( ) Model Repository (truth)
2 ( ) Data Access Engine (spin)
3 ( ) Command Engine (way)
```

```
4 ( ) Software Repository (word)
5 ( ) Opsware Global Filesystem Server (OGFS)
6 ( ) Opsware Documentation
7 ( ) Opsware Command Center (OCC)
8 ( ) OS Provisioning Media Server
9 ( ) OS Provisioning Build Manager
10 ( ) Opsware Gateway
11 ( ) OS Provisioning Boot Server

Enter a component number to toggle ('a' for all, 'n' for
none).
When ready, press 'c' to continue, or 'q' to quit.

Selection:
```

You must install the components in the order they are listed. For example, you must install the Model Repository before the Data Access Engine.

If you are installing all of the components on a single server, then you may enter `a` for all. If you do not select `a`, then you must run the Opsware Installer again (specifying the response file) and select the remaining components. (If you are installing the components on multiple servers, see the next step.)

For some of the components, such as the OS Provisioning Build Manager, the Installer interview prompts you for the IP address or host name. Be sure to install these components on the host that you indicated during the interview.

**14** If you are installing the components on multiple servers, follow the instructions in this step. (If you are installing the components on a single server, skip this step.)

- Copy the response file generated by the installer interview to all other servers in this core.
- After you install the Model Repository, copy the Oracle `tnsnames.ora` file from the server with the Model Repository to the other Opsware core servers. The directory path for the file must be the same on all core servers. (By default, `tnsnames.ora` is in the `/var/opt/oracle` directory.)
- On each server in this core, run the Opsware Installer with the `-r` option, as shown in step 12. Select and install the remaining components from the menu shown in step 13.
- For the Model Repository, the installer asks if you want to generate cryptographic material, enter `y`. Copy the database of cryptographic material from the following directory to every Opsware core server:

`/var/lc/crypto/cadb/realm/opsware-crypto.db.e`

The database of cryptographic material must be copied to the same directory and file name on every Opsware core server. The directory and database need to be readable by the root user.

- You must install the Opsware Documentation component on the server where you install the Opsware Command Center component.
- If the Model Repository or Boot Server exist on a server with no other Opsware components installed on it, you must install an Opsware Agent on that server. See the *Opsware® SAS User's Guide* for instructions.

**15** (Optional) If you are distributing the core components across multiple servers, you may install additional instances of the following components:

- Data Access Engine

  If you install more than one Data Access Engine, then you must perform the procedure described in "Reassigning the Data Access Engine to a Secondary Role" in the *Opsware® SAS Administration Guide*.
- OS Provisioning Media Server
- Opsware Command Center
- Opsware Global File System Server

**16** Follow the instructions in the following section, "Opsware Command Center Web Client" on page 70

**17** Follow the instructions in "Post-Installation Tasks" on page 73.

## Opsware Command Center Web Client

After you install an Opsware SAS core, you should be able log into the Opsware Command Center web client.

To run the Opsware Command Center, your browser must be configured in the following manner:

- The browser must accept cookies and be able to use Java.
- The browser must support SSL and should provide 128-bit encryption (recommended).
- Using a pop-up blocker might prevent some functions from working correctly. Either disable the pop-up blocker completely or use the supported browser's native pop-up blocking function instead of a third-party product.

## Logging into the Opsware Command Center

1. In a web browser, enter the following URL:

   `https://<ip-address-occ>`

   The `<ip-address-occ>` is the IP address of the server on which you installed the Opsware Command Center component.

2. Follow the browser's instructions for installing the security certificate.

3. When the Opsware Command Center prompts you for the user name and password, enter `admin` for the user name. For the password, enter the value for the `cast.admin_pwd`, which you specified during the Installer interview.

4. Create a new user by using the Users & Groups page under Administration. For the Group Membership, select Opsware System Administrators.

   See the *Opsware® SAS Configuration Guide* for information about creating Opsware users.

5. Log into the Opsware Command Center as the user you created in the previous step. Run the Opsware System Diagnosis by clicking System Diagnosis under Administration in the navigation panel.

   See the *Opsware® SAS Administration Guide* for information about the procedures for running the system diagnosis tool.

6. Log into the Opsware Command Center as the `admin` user again. Create a new user and for the Group Membership, select Advanced Users.

7. Log into the Opsware Command Center as the user you created in the previous step. Exercise the different Opsware System functions by clicking the links in the left navigation panel and by opening the wizards on the home page.

# Chapter 5: Post-Installation Tasks

**IN THIS CHAPTER**

This section discusses the following topics:

- Setup for Opsware Discovery and Deployment
- DHCP Configuration for OS Provisioning
- Additional Network Requirements for OS Provisioning
- Patch Management on Windows NT 4.0 and Windows 2000

## Setup for Opsware Discovery and Deployment

With the Opsware Discovery and Deployment (ODAD) feature you can use the OCC Client to install Opsware Agents on servers.

### Installing the Windows Agent Deployment Helper

Before using the ODAD feature to install Agents on Windows servers, you must install the Windows Agent Deployment Helper package.

You need to install only one Windows Agent Deployment Helper for each Opsware core. You cannot install a Windows Agent Deployment Helper in an Opsware Satellite.

To install the Windows Agent Deployment Helper, perform the following steps:

1. Obtain a Windows server on which you can install the Windows Agent Deployment Helper. On this Windows server, install an Opsware Agent with the command-line utility. See the *Opsware® SAS User's Guide* for instructions on how to install an Opsware Agent with the command-line utility.
2. If the Administrator account on the Windows server has been renamed, then create an account named Administrator (with administrator privileges).

3. Log into the Opsware Command Center as a member of the Opsware System Administrator group.
4. From the navigation panel, select Servers ➤ Manage Servers. The Manage Server page appears. Select the Windows server where you just installed the Opsware Agent.
5. From the **Tasks** Menu, select **Install by Templates**. The Install Template page appears.
6. Navigate to Opsware Tools ➤ Agent Deployment Helper ➤ Windows *version*
7. Click **Next**. The Confirm Selections page appears.
8. Review your selections and install the template.

   The Windows Agent Deployment Helper is installed and the server is attached to this node: Service Level ➤ Opsware ➤ adh ➤ windows.
9. Restart any running OCC Clients.

   The restart is needed because the OCC Client caches information about the Windows Agent Deployment Helper.
10. Log in as `root` onto the server with the core Gateway. With a text editor, open the following file:

    `/var/opt/OPSWgw/cgw0-<facility>/opswgw.properties`
11. Locate the following line:

    `#opswgw.IngressMap=${NETBIOSHELPERIP}:NETBIOS`
12. Uncomment the line, and replace `${NETBIOSHELPERIP}` with the IP address of the server where you installed the Windows Agent Deployment Helper. For example:

    `opswgw.IngressMap=192.168.165.242:NETBIOS`
13. Restart the core Gateway with the following command:

    `/etc/init.d/opswgw-cgw0-<facility> restart`

### ODAD for Unix Servers

Before using the ODAD feature to open remote terminal sessions on unmanaged Unix servers, verify the following requirement. On the server with the Agent Gateway, the `telnet`, `rlogin`, and `ssh` clients must reside in either the `/bin`, `/usr/bin`, or `/usr/local/bin` directory. If the client resides in a different directory, create a symbolic link in `/usr/local/bin` to the actual location of the client.

## DHCP Configuration for OS Provisioning

The Dynamic Host Configuration Protocol (DHCP) specifies how to assign dynamic IP addresses to servers on a network. Opsware OS Provisioning uses DHCP to allow network booting and configuration of unprovisioned servers in the Server Pool. DHCP is also used to configure networking on newly provisioned servers that have not been assigned a static network configuration.

For OS provisioning, you may use either the DHCP server included Opsware SAS, an existing ISC DHCP server, or the MS Windows DHCP server. The instructions for configuring these various DHCP servers are in the following sections:

- Configuring the Opsware DHCP Server for OS Provisioning
- Configuring an Existing ISC DHCP Server for OS Provisioning
- Configuring the MS Windows DHCP Server for OS Provisioning
- Configuring the Opsware and MS Windows DHCP Servers for OS Provisioning

### DHCP Software included with the Opsware Boot Server

When you install the Opsware Boot Server, the Opsware Installer also installs the following items:

- **dhcpd**: An Internet Software Consortium DHCP server (ISC dhcpd).
- **dhcpd.conf**: A default configuration file, read by the `dhcpd` server.
- **dhcpdtool**: The Opsware DHCP Network Configuration Tool, writes to the `dhcpd.conf` file.

#### *Opsware DHCP Server (dhcpd)*

The DHCP server provides service to two types of networks:

- **Local networks**: Networks that are attached directly to the network interfaces of the host running the DHCP server. No special network configuration is needed to support local networks.
- **Remote networks**: Networks that are not directly attached to the DHCP server host. A router sits between the DHCP server host and the remote networks. For remote networks, a DHCP proxy (sometimes called IP helper) must be configured on each remote network to relay DHCP packets to the DHCP server host.

A DHCP proxy is not provided with Opsware SAS and instructions for setting one up are beyond the scope of this document. DHCP proxy functionality is often included in modern routers. Check with your network administrator or router vendor.

Log messages that the DHCP server produces are sent to the standard Unix syslog process with the daemon facility. Consult your vendor documentation on how to configure and view syslog messages.

See "Starting and Stopping the Opsware DHCP Server" on page 80.

### *Opsware dhcpd.conf File*

The `dhcpd.conf` file provides the necessary parameters to support network booting of Sun hardware (a DHCP-capable PROM is required) and x86 hardware (a PXE-compatible system is required).

For x86 hardware that does not support PXE, the server can be booted from a floppy (Windows) or CD (Linux). When a boot floppy or CD is used, the DHCP server still provides network configuration information to the host.

The DHCP configuration file is `/opt/OPSWdhcpd/etc/dhcpd.conf`. In most cases, you will modify this file by running the DHCP Network Configuration Tool. For some advanced configurations (as noted in the following section), you may need to modify the file with a text editor. Documentation on the DHCP configuration file is available at the ISC web site `www.isc.org`.

The DHCP leases file is `/var/opt/OPSWdhcpd/dhcpd.leases`. Normally, this file should not need editing.

### *Opsware DHCP Network Configuration Tool (dhcpdtool)*

The DHCP Network Configuration Tool is a menu-driven, terminal-based utility that enables you to customize the `dhcpd.conf` file for common local and remote network configurations. The tool prompts you for network information needed to configure DHCP for each OS provisioning network. Using the DHCP Network Configuration Tool simplifies configuration of the DHCP server and ensures that the DHCP configuration contains the options that are needed for the OS Provisioning feature to function properly.

If you need to configure the network for Opsware OS Provisioning to support less common configurations, you must modify the `dhcpd.conf` file with a text editor. Less common configurations include dual-interfaces with split-horizon DNS requirements, private build networks, and static NAT. Contact Opsware Support for more assistance.

Additionally, in some environments, multiple IP networks (layer 3) are layered on top of a single VLAN (layer 2). While this configuration is supported by the ISC DHCP server, generally such a topology requires careful consideration to work properly with DHCP. Therefore, the DHCP Network Configuration Tool can only configure a single IP network per VLAN.

The man pages for the DHCP Network Configuration Tool are installed in `/opt/OPSWdhcpd/man` on the Boot Server. They are also available at the Opsware Support web site.

### *Required Information for the Opsware DHCP Network Configuration Tool*

Before you use the DHCP Network Configuration Tool to configure an OS provisioning network, you need the following information:

- The range of IP addresses that are assigned dynamically by the DHCP server. For example, 192.168.0.11, 192.168.0.20 might be used to configure a pool of 10 addresses. **Important**: Each of these IP addresses must resolve to a host name on the DNS server.
- The IP addresses of one or more DNS servers. The servers given must be able to resolve the standard required Opsware DNS entries. The DNS servers do not need to be on the same network that is being configured.
- A default DNS domain. This domain must include the standard, required Opsware DNS entries. For example, if the default DNS domain is `example.org`, then there must be an entry `spin.example.org` that can be resolved by the DNS servers.

If you are going to configure a remote network with the DHCP Network Configuration Tool, you will also need to provide the following information:

- The network address and size (netmask or bits). For example, 192.168.0.0/255.255.255.0 or 192.168.0.0/24. Both specify a network range of 192.168.0.0 - 192.168.0.255.
- The network gateway or default router, for example, 192.168.0.1.

### Configuring the Opsware DHCP Server for OS Provisioning

The DHCP Network Configuration Tool is installed with the Opsware Boot Server. Perform the following steps to configure networks for OS provisioning:

1. Log in as root to the server running the Opsware Boot Server.
2. Make a backup copy of the configuration file:

   ```
   cd /opt/OPSWdhcpd/etc
   cp dhcpd.conf dhcpd.conf.orig
   ```

3. Run the DHCP Network Configuration Tool:

   ```
   /opt/OPSWdhcpd/sbin/dhcpdtool
   ```

   The DHCP Network Configuration Tool main menu appears, as follows:

**Example: DHCP Network Configuration Tool Main Menu**

```
Opsware DHCP Network Configuration Tool

 a)dd a new network.
 e)xit.

Choice [a, e]:
```

4. To add a new network, enter `a` at the preceding prompt.

   The menu to add local or remote networks appears, as follows.

**Example: Menu to Add Local or Remote Networks**

```
Opsware DHCP Network Configuration Tool

You may view/edit/delete one of the currently configured
network(s):

 1) 192.168.164.0/28
 2) 192.168.165.128/28
```

```
Or

 a)dd a new network.
 e)xit.

Choice [1..2, a, e]: a:
```

5. To configure the DHCP service on the local network, enter `l` at the preceding prompt. Local networks are detected automatically and displayed.

   Or

   To add a remote network, enter `r` at the preceding prompt.

6. If you are adding a local network, you need to enter the IP addresses or host names of the DHCP range and the DNS servers. In the example that follows, note that the IP addresses are separated by a comma and a space.

**Example: Local Network Configuration**

```
Opsware DHCP Network Configuration Tool

Editing DHCP information for 192.168.8.0/23 (255.255.254.0)

All values which prompt for an address accept either a IP or a
hostname.

Enter the DHCP Range (start address, stop address)
: 192.168.8.20, 192.168.8.29
Enter the DNS server(s) (comma separated)
: 192.168.2.25, 192.168.2.28
Enter the DNS domain: opsware.com
```

7. If you are adding a remote network, you need to supply information for the network address, size, and gateway. See the example that follows.

**Example: Remote Network Configuration**

```
Opsware DHCP Network Configuration Tool

All values which prompt for an address accept either a IP or a
hostname.

Enter network/netmask or network/bits: 192.168.10.0/24
Enter the network gateway: 192.168.10.1
Enter the DHCP Range (start address, stop address)
```

```
: 192.168.10.51, 192.168.10.59
Enter the DNS server(s) (comma separated)
: 192.168.2.25, 192.168.2.28
Enter the DNS domain: opsware.com
```

8. If the displayed information is correct, enter `k` to keep the network and return to the main menu.
9. At the main menu, to save the information you have entered, enter `s`.

   Or

   To edit a configured network, enter the corresponding integer and go back to step 3.

   Or

   To add more networks, enter `a` and go back to step 3.
10. To exit the DHCP Network Configuration Tool, enter `e`. You are prompted to start (or restart) the DHCP server process.
11. To start (or restart) the DHCP server process, enter `y`. The DHCP Network Configuration Tool displays diagnostic output as part of its startup.

### Starting and Stopping the Opsware DHCP Server

To start the DHCP server process, enter the following command on the server running the Opsware Boot Server:

```
/etc/init.d/dhcpd start
```

To stop the DHCP server process, enter the following command on the server running the Opsware Boot Server:

```
/etc/init.d/dhcpd stop
```

### Configuring an Existing ISC DHCP Server for OS Provisioning

You may use an existing ISC DHCP server for OS provisioning instead of the DHCP server included with Opsware SAS. An existing ISC DHCP server will work with the provisioning of PXE 2.0 clients, but not with older clients such as PXE 0.99 or 1.0. (These older PXE clients have old PROMS and a PXE bootstrap floppy made with `rbfg.exe`.) The instructions that follow apply to recent versions of an ISC DHCP server, such as version 3.02rc3.

To configure an existing ISC DHCP server, perform the following steps:

1. On the server where you installed the Opsware Boot Server, you should prevent the Opsware DHCP server from running.

   On Linux, enter the following command:

   ```
   chkconfig --level 345 dhcpd off
   ```

   On Solaris, enter these commands:

   ```
   rm /etc/rc2.d/S90dhcpd
   rm /etc/rc0.d/K30dhcpd
   ```

2. Ensure that the configuration file for the existing ISC DHCP server has the entries shown in: "Example: Configuration File Entries for an Existing ISC DHCP Server" on page 81.

   The example is a snippet of the `dhcp.conf` shipped with Opsware SAS, with the addition of `next-server`. This addition tells the PXE client to look for the `tftpserver` on the Opsware core, not on the existing DHCP server.

3. If you copy and paste the example, change all of the IP addresses (`1.2.3.4`) to the IP address of your core.

4. Make sure the that the DHCP scope for the systems to be provisioned is set up with the required details: DNS server, netmask, default router, DNS domain, and so forth.

5. Restart the existing ISC DHCP server.

### *Example: Configuration File Entries for an Existing ISC DHCP Server*

```
#
# declare OPSW site options
#
option space OPSW;
# DANGER WILL ROBINSON - if you change the codes for these
options,
# you'll need to also edit them in the param-request-lists
appearing
# below. Note that in the pxeclient section, you need to specify
the
# values in hex, not in decimal. Also, these values are burned
into
# a couple other files you'll need to edit as well:
# /opt/OPSWboot/tftpboot/pxelinux.cfg/default
# /opt/OPSWboot/jumpstart/Boot/etc/dhcp/inittab
# /opt/OPSWboot/jumpstart/Boot/etc/default/dhcpagent
option OPSW.buildmgr_ip code   186 = ip-address;
option OPSW.buildmgr_port code 187 = unsigned integer 16;

#
```

```
# define OPSW site options
#
site-option-space "OPSW";
option OPSW.buildmgr_ip   1.2.3.4;
 option OPSW.buildmgr_port 8017;

#
# declare SUNW jumpstart vendor options (Sun recommended naming)
#
option space SUNW;
option SUNW.SrootIP4 code 2  = ip-address;
option SUNW.SrootNM  code 3  = text;
option SUNW.SrootPTH code 4  = text;
option SUNW.SbootFIL code 7  = text;
option SUNW.SinstIP4 code 10 = ip-address;
option SUNW.SinstNM  code 11 = text;
option SUNW.SinstPTH code 12 = text;
option SUNW.SsysidCF code 13 = text;
option SUNW.SjumpsCF code 14 = text;
option SUNW.Sterm    code 15 = text;

#
# define SUNW jumpstart vendor options
#
class "solaris-sun4u" {
   match option vendor-class-identifier;
   vendor-option-space SUNW;
   next-server 1.2.3.4;
   option SUNW.SrootIP4 1.2.3.4;
   option SUNW.SrootNM  "js";
   option SUNW.SrootPTH "/opt/OPSWboot/jumpstart/Boot";
   option SUNW.SinstIP4 1.2.3.4;
   option SUNW.SinstNM  "js";
   option SUNW.SjumpsCF "js:/opt/OPSWboot/jumpstart/Conf";
   option SUNW.SsysidCF "js:/opt/OPSWboot/jumpstart/Conf";
   option SUNW.Sterm    "vt100";
   option SUNW.SbootFIL "/platform/sun4u/kernel/sparcv9/unix";
   # We use a bogus install path just to give the installer
something to
   # mount for now.
   option SUNW.SinstPTH "/opt/OPSWboot/jumpstart/Boot";
   option dhcp-parameter-request-list 1,3,6,12,15,43,186,187;
}

# Begin dhcptool added SUNW client classes (do not edit)
subclass "solaris-sun4u" "FJSV.GPUU";
subclass "solaris-sun4u" "NATE.s-Note_737S";
```

```
subclass "solaris-sun4u" "NATE.s-Note_747S";
subclass "solaris-sun4u" "NATE.s-Note_777S";
subclass "solaris-sun4u" "SUNW.Netra-T12";
subclass "solaris-sun4u" "SUNW.Netra-T4";
subclass "solaris-sun4u" "SUNW.Sun-Blade-100";
subclass "solaris-sun4u" "SUNW.Sun-Blade-1000";
subclass "solaris-sun4u" "SUNW.Sun-Fire-15000";
subclass "solaris-sun4u" "SUNW.Sun-Fire-280R";
subclass "solaris-sun4u" "SUNW.Sun-Fire-480R";
subclass "solaris-sun4u" "SUNW.Sun-Fire-880";
subclass "solaris-sun4u" "SUNW.Sun-Fire";
subclass "solaris-sun4u" "SUNW.Ultra-1-Engine";
subclass "solaris-sun4u" "SUNW.Ultra-1";
subclass "solaris-sun4u" "SUNW.Ultra-2";
subclass "solaris-sun4u" "SUNW.Ultra-250";
subclass "solaris-sun4u" "SUNW.Ultra-30";
subclass "solaris-sun4u" "SUNW.Ultra-4";
subclass "solaris-sun4u" "SUNW.Ultra-5_10";
subclass "solaris-sun4u" "SUNW.Ultra-60";
subclass "solaris-sun4u" "SUNW.Ultra-80";
subclass "solaris-sun4u" "SUNW.Ultra-Enterprise-10000";
subclass "solaris-sun4u" "SUNW.Ultra-Enterprise";
subclass "solaris-sun4u" "SUNW.UltraAX-MP";
subclass "solaris-sun4u" "SUNW.UltraAX-e";
subclass "solaris-sun4u" "SUNW.UltraAX-e2";
subclass "solaris-sun4u" "SUNW.UltraAX-i2";
subclass "solaris-sun4u" "SUNW.UltraSPARC-IIe-NetraCT-40";
subclass "solaris-sun4u" "SUNW.UltraSPARC-IIe-NetraCT-60";
subclass "solaris-sun4u" "SUNW.UltraSPARC-IIi-Engine";
subclass "solaris-sun4u" "SUNW.UltraSPARC-IIi-Netract";
subclass "solaris-sun4u" "SUNW.UltraSPARC-IIi-cEngine";
subclass "solaris-sun4u" "SUNW.UltraSPARCengine_CP-20";
subclass "solaris-sun4u" "SUNW.UltraSPARCengine_CP-40";
subclass "solaris-sun4u" "SUNW.UltraSPARCengine_CP-60";
subclass "solaris-sun4u" "SUNW.UltraSPARCengine_CP-80";
# End dhcptool added SUNW client classes (do not edit)

#
# declare PXE vendor options
#
option space PXE;
option PXE.mtftp-ip             code 1  = ip-address;
option PXE.mtftp-cport          code 2  = unsigned integer 16;
option PXE.mtftp-sport          code 3  = unsigned integer 16;
option PXE.mtftp-tmout          code 4  = unsigned integer 8;
option PXE.mtftp-delay          code 5  = unsigned integer 8;
option PXE.discovery-control    code 6  = unsigned integer 8;
```

```
option PXE.discovery-mcast-addr code 7  = ip-address;
option PXE.boot-item            code 71 = unsigned integer 16;

#
# define PXE vendor options
#
class "pxeclients" {
   match if substring (option vendor-class-identifier, 0, 9) =
"PXEClient";
   vendor-option-space PXE;
   filename "pxelinux.0";
   next-server 1.2.3.4;
   option vendor-class-identifier "PXEClient";
   # We set the MCAST IP address to 0.0.0.0 to tell the boot ROM
we
   # can't provide multicast TFTP, so it will have to use just
   # plain ol' TFTP instead (address 0.0.0.0 is considered
   # as "no address").
   option PXE.mtftp-ip 0.0.0.0;
   option dhcp-parameter-request-list = concat(dhcp-parameter-
request-list,ba,bb);
}
```

### Configuring the MS Windows DHCP Server for OS Provisioning

You may use the MS Windows DHCP server instead of the Opsware DHCP server to provision Windows or Linux on PXE 2.0 clients. The MS Windows DHCP server cannot be used during the OS provisioning of the following types of systems:

- Solaris
- PXE 0.99, 1.x clients (These older PXE clients have old PROMS and a PXE bootstrap floppy made with `rbfg.exe`.)

To configure the MS Windows DHCP server for OS Provisioning, perform the following steps:

1. On the MS Windows system running the DHCP server, you need to add the option #60, so that it appears in the DHCP scope options. Open a command prompt, and enter the following command:

   ```
   netsh.exe dhcp server add optiondef 60 "PXEClient" STRING
   ```

2. Using the DHCP management snap-in (`dhcpmgmt.msc`), create a scope, which is usually a subnet declaration. In the scope options, #60 should now appear. Check the box, and then add the string `PXEClient`.

3. Using the same scope options box, configure options 66 and 67: Click the DHCP option #66 (Boot Server Host Name), and add the full DNS name of the tftp/boot server (for example `core01.test.com`). For option #67 (Bootfile Name), add the boot file name: `pxelinux.0`.

4. Make sure the that the DHCP scope for the systems to be provisioned is set up with the required details: DNS server, netmask, default router, DNS domain, and so forth.

5. At the command prompt, enter the following commands to locate the IP address of the Opsware Agent Gateway and the port forward for the Build Manager:

```
netsh.exe dhcp server add optiondef 186 "buildmgr_ip" IPADDRESS
netsh.exe dhcp server add optiondef 187 "buildmgr_port" WORD
```

6. Using the DHCP management snap-in (`dhcpmgmt.msc`), configure the options 186 and 187 to be part of your scope, and give them the appropriate values (IP address of the Opsware Agent Gateway and the port forward for the Build Manager, normally 8017).

7. Also in the scope, define option 043 (Vendor specific options) as a BINARY type, with the value `01 04 00 00 00 00 ff`. This setting tells the DHCP server to go directly to the tftp server specified in the Boot Server Host Name parameter, and also tells it to not use Multicast TFTP.

8. Restart the MS Windows DHCP server.

### Configuring the Opsware and MS Windows DHCP Servers for OS Provisioning

You can configure the Opsware DHCP server to respond only to the OS provisioning requests (that is, from the PXE and Solaris clients), while the MS Windows DHCP server responds to all other requests.

1. Add the network subnet to the Opsware DHCP server. See “Configuring the Opsware DHCP Server for OS Provisioning” on page 78.

2. Stop the Opsware DHCP server:

```
/etc/init.d/dhcpd stop
```

3. Make a copy of the Opsware DHCP configuration file:

```
cd /opt/OPSWdhcpd/etc
cp dhcpd.conf dhcpd.conf.orig
```

4. In a text editor, open the Opsware DHCP configuration file.

5. In the text editor, find the subnet definition you want to configure and comment out (with the # character) these lines:

   ```
   range <IP1> <IP2>;
   ```

6. Immediately after the commented out line (# range), enter lines such as:

   ```
   pool {
           allow members of "solaris-sun4u";
           allow members of "solaris-sun4us";
           allow members of "pxeclients";
           range <IP1> <IP2>;
           }
   ```

   The preceding `pool` statement tells the DHCP server to continue serving the range specified, but only for the three types of clients indicated. (The first two `allow` statements are for Sun machines, the third is for PXE clients). In the preceding `pool` statement, be sure to include the closing brace }.

7. Repeat the preceding two steps for every subnet you wish to configure.

8. In the text editor, save the `dhcpd.conf` file.

9. Start the Opsware DHCP server:

   ```
   /etc/init.d/dhcpd start
   ```

10. Check the following logs for DHCP errors:
    `/var/log/messages`
    `/var/adm/messages`

11. Make sure that the MS Windows DHCP server subnet/scope declarations are changed to include the build manager DHCP options (code 186 and 187). See “Configuring the MS Windows DHCP Server for OS Provisioning” on page 84.

12. Make sure that the MS Windows DHCP server does not include options 43, 60, 66, or 67 in the scope/subnets you are configuring. This will prevent the PXE and Sun jumpstart clients from talking to the MS Windows DHCP server. Instead, they will talk to the Opsware DHCP server.

13. Make sure that the IP ranges of the MS Windows and Opsware DHCP servers don't overlap. As a guideline, the number of IP addresses in a given range should be twice the maximum number of servers that will be provisioned concurrently.

14 If the DHCP servers aren't directly connected to the network/subnet of the systems being provisioned, the DHCP requests must be forwarded to both DHCP servers, with the Opsware DHCP server being first.

## Additional Network Requirements for OS Provisioning

### *OS Provisioning for Solaris*

If you are using OS provisioning for Solaris (JumpStart) on an isolated network, you must have a default gateway (router) available, even if it does not route packets. For Solaris JumpStart to function properly, the IP address of the default gateway must be sent to the installation client that is being provisioned with DHCP. When you use the Opsware DHCP Configuration Tool, a default gateway is properly configured for Solaris because the DHCP Configuration Tool adds the default router appropriately.

### *Host Name Resolution*

For Windows OS provisioning, the host name `buildmgr` should resolve on Windows installation clients.

The Opsware core host names must resolve using the DNS search order and DNS server information that the DHCP server provides. The DHCP server provides the DNS server IP address and the DNS search order. For each subnet you configure with the Opsware DHCP Tool, the DNS domain used by that subnet must have a DNS entry for `buildmgr`.

For example, you could have two subnets with the following domain names:

```
subnet1.example.com
subnet2.example.com.
```

Therefore, there must be two DNS entries:

```
buildmgr.subnet1.example.com
buildmgr.subnet2.example.com.
```

The host running the OS Provisioning Media Server must be able to resolve the IP address to the host name (a reverse lookup) of a server being provisioned.

See also “Host and Service Name Resolution Requirements” on page 34.

### *Open Ports*

The server on which the OS is to be provisioned has the same requirements for connectivity to the Opsware core network as a managed server. See “Open Ports” on page 32.

## Patch Management on Windows NT 4.0 and Windows 2000

To use the `mbsacli.exe` patch utility for patch management on Windows NT 4.0, you must first install Internet Explorer 6.0 or later because the `mbsacli.exe` patch utility depends on it. This prerequisite is not required for Windows 2003 because IE 6.0 is pre-installed for this operating system.

### Creating a Silent Installable Version of IE 6.0 or Later

To create a silent-installable version of IE 6.0 or later, use the Internet Explorer Administrator's Kit (IEAK) for the version of IE that you want to install. For more information on IEAK, see the following URL:

`http://microsoft.com/windows/ieak/default.asp`

Perform the following steps to create a silent installable version of IE 6.0 or later:

1. Install IEAK on your desktop system.
2. After you install IEAK, start the Internet Explorer Customization Wizard.
3. When creating the package, IEAK prompts for a Media Selection option. Select the option Flat (all files in one directory).
4. Select the defaults for all other options when you use the wizard.
5. After the wizard is complete, zip the contents of the directory it created. This directory contains the silent-installable version of IE.
6. To upload the ZIP package into Opsware SAS, click Software ➤ Packages in the Opsware Command Center left navigation panel, then select the Upload tab in the Browse Packages page, and then follow the prompts to upload the package.
7. Set the following attributes for the package when you upload it into Opsware SAS:
   - In the Customer field, select Customer Independent.
   - In the Operating System field, select Windows NT 4.0 or Windows 2000, depending on the version of Windows for which you are setting up silent installation.
   - In the Package Type field, select Windows Zip file.
   - In the Edit Properties page in the Installation Directory field, enter the installation location:

     `%SystemDrive%\IE-redist`

- In the Edit Properties page in the Post-Install Script Filename field, enter this text:

  `%SystemDrive%\IE-redist\ie5setup.exe /q:a /r:n`

  Where `ie6setup.exe` is the IE 6.x stub installer

  The /q:a install option specifies quiet install mode, with no user prompts. The /r:n install option suppresses restarting the server after IE installation.

- In the Edit Properties page in the Reboot on Successful Install field, select the Yes option.

8 Create a node in a Software category or in Templates and attach the uploaded packages to the node.

9 Use the Install Software Wizard of the Opsware Command Center to install the necessary software on a Windows NT 4.0 managed server.

# Chapter 6: Opsware Multimaster Installation

**IN THIS CHAPTER**

This section discusses the following topics:

- Multimaster Installation
- Components of Multimaster Installations
- Converting a Core from Standalone to Multimaster
- Adding a Core to a Multimaster Mesh
- Multimaster Post-Installation Tasks

## Multimaster Installation

An Opsware multimaster mesh contains two or more cores that communicate with each other. This section refers to the first core you install in a multimaster mesh as the source core. The target core is the second, third, or subsequent core that you install in a multimaster mesh. (In some prompts, the Opsware Installer uses the term slave core instead of target core.)

The main phases in creating a multimaster mesh of cores are shown in the following steps:

1. Install a standalone (source) core.
   - Run the Opsware Installer interview, saving the data you enter at the prompts in a response file.
   - Run the installer again, specifying the response file, on one or more servers to install the Opsware components.
   - See “Installing a Standalone Core” on page 66.
2. Convert the standalone core to a multimaster core.
   - Run the Opsware Installer interview with the response file created in the previous step, and then save your answers for this interview in another response file.

- Run the installer again, specifying the latest response file, on one or more servers to add the multimaster components to the source core.
- See “Converting a Core from Standalone to Multimaster” on page 94.

3 Add the new target core to the multimaster mesh.

- On the source core, run the Opsware Installer interview with the response file created in the previous step, and then save your answers for this interview in another response file.
- Run the installer again, specifying the latest response file, and instruct the installer to define a new facility.
- Run the installer again to export data from the Model Repository and to create a global response file.
- Copy the export data file and the global response file from the source core server to the target core server.
- On the target core, run the Opsware Installer interview with the global response file and save your answers for this interview in another response file.
- Run the installer again, specifying the latest response file, on one or more servers to install the components of the target core.
- See “Adding a Core to a Multimaster Mesh” on page 98.

For a given multimaster mesh, you perform steps 1 and 2 one time only. You perform step 3 every time you want to add another core to the multimaster mesh.

## Components of Multimaster Installations

This section discusses the following topics:

- Pre-Existing Core Installations
- Opsware Command Center
- Prerequisites for a Multimaster Installation
- TIBCO Rendezvous

### Pre-Existing Core Installations

If you installed a standalone core at any secondary facilities and you want to include these facilities in your multimaster mesh, you must perform the following tasks:

- Uninstall the Opsware core at the secondary facilities. See "Opsware SAS Uninstallation" on page 135 in Chapter 9 for more information.
- Follow the instructions in the section "Multimaster Installation" on page 91.

### Opsware Command Center

Target facilities (cores) in the multimaster mesh are not required to have an Opsware Command Center installed. Instead, you can manage the facility from any site in the multimaster mesh that does have an Opsware Command Center installed. You need to install the Opsware Command Center only if you want to manage your multimaster mesh locally from that facility or if you want to have a backup Opsware Command Center.

### TIBCO Rendezvous

In a multimaster mesh, Opsware SAS uses the TIBCO Certified Messaging system to synchronize Model Repositories at different facilities.

When you add a core to a multimaster mesh, the Opsware Installer automatically configures the TIBCO Rendezvous routing daemon (`rvrd`). For more information, see "TIBCO Rendezvous Configuration for Multimaster" on page 151.

### Prerequisites for a Multimaster Installation

Perform the following tasks in preparation for installing a multimaster core:

- Plan your Opsware System deployment. When planning for a core, you must decide whether you want to install the core components on a single server or on multiple servers. See the *Planning Deployments for Opsware® SAS*.
- Perform the pre-installation administration tasks such as configuring the network. See "Pre-Installation Requirements" on page 29.
- Create an Oracle database, which is required for the Opsware Model Repository. See "Oracle Setup for Model Repository" on page 141.
- Gather information in preparation for the Opsware Installer interview. This information includes the name and ID of the facility for the core. See "Prerequisite Information for the Installer Interview" on page 41.

- Verify that every Opsware core server has a unique IP address within the entire multimaster mesh.
- After you synchronize the time on all servers within a facility, synchronize the time between the facilities in the multimaster mesh. Synchronize the time with an external time-server that uses Network Time Protocol (NTP) so that all servers are using the same Coordinated Universal Time (UTC).
- Verify that the multimaster installation meets same network requirements as a standalone installation, except that each core must be on a different Local Area Network (LAN or VLAN). The cores must be in different broadcast domains.
- Make sure that each core in a mesh has a different subdomain so that managed servers can resolve the unqualified host names `spin`, `way`, and `theword`.
- Verify that the `tnsnames.ora` file on the source core contain entries for every Model Repository in the mesh. If the `tnsnames.ora` file of the source core does not contain an entry for the target core, then multimaster conflicts will occur. The entries in the `tnsnames.ora` file are different in version 5 of Opsware SAS than in version 4. For more information, see "Set up the tnsnames.ora file." on page 143.

## Converting a Core from Standalone to Multimaster

This section describes how to convert an Opsware core from standalone to multimaster. Throughout this section, the core to be converted is referred to as the source core. (If you already have a multimaster mesh and want to add an additional core, go to the section "Adding a Core to a Multimaster Mesh" on page 98.)

Perform the following steps to convert a core from standalone to multimaster:

1. Obtain the Opsware SAS installation media for this release.

   See "Installation Media for the Opsware Installer" on page 62, including the recommendation, "Copying the DVD to a Local Disk."

2. On each server of the source core, mount the DVD or NFS-mount the directory that contains a copy of the DVD contents.

   The Opsware Installer must have read/write root access to the directories where it installs Opsware components, even NFS-mounted network appliances.

3. On the Model Repository server in the source core, log in as root.
4. Change to the root directory:

```
cd /
```

5 invoke the Opsware Installer with the `-r` (response file) and the `--interview` options. For example:

```
/opsware_system/opsware_installer/install_opsware.sh -r /
usr/tmp/oiresponse.stand_single --interview
```

You must specify the full path to the script. The directory path in the preceding command indicates that you copied the Opsware SAS DVD to a local disk or network share using the required directory structure.

You should run the Opsware Installer with the response file that you created when you installed the source core. If this response file is not available, invoke the Opsware Installer with no command line options, and the interview will automatically start.

The Opsware Installer displays the following options:

```
Welcome to the Opsware Installer. Please select one of the
following installation options:

1 - Standalone Installation: Standalone Opsware Core
2 - Multimaster Installation: First Core (convert from
standalone)
3 - Multimaster Installation: Define New Facility; Export
Model Repository
4 - Multimaster Installation: Additional Core
5 - Satellite Installation: Opsware Satellite
```

6 At the installation options prompt, select the following option:

```
2 - Multimaster Installation: First Core (convert from
standalone)
```

7 At the interview mode prompt, select one of the following options:

```
1 - Simple Interview Mode
2 - Advanced Interview Mode
```

Option 1 is for using default values for many of the configuration parameters. Option 2 is for specifying all configuration parameters during the interview.

8 Respond to the interview prompts.

The installer displays default values in square brackets [ ].

See “Required Information for Running the Installer Interview” on page 41.

9 Decide if you want to finish the interview.

When you enter all of the required information, the Opsware Installer displays this message:

```
All parameters have values. Do you wish to finish the
interview (y/n):
```

If you are satisfied with your answers, press y.

If you want to review or change your answers, press n. The installer displays the prompts again, showing in brackets [ ] the values that you previously entered.

**10** Create the response file.

When you are finished with the interview, the installer prompts you for the name of the response file:

```
Name of response file to write
[/usr/tmp/oiresponse.stand_to_mm]
```

The response file is a text file that contains the answers you entered during the interview. You can enter the name of the response file or accept the default. In either case, write down the name of the response file. Note that the default file name corresponds to the type of installation.

**11** The Opsware Installer prompts you to indicate whether you want to continue the installation by using the response file. Select one of the following options:

- If you are satisfied with the responses you entered in the interview and you are ready to install the Model Repository Multimaster Additions now, enter `y` to continue.
- If you do not want to install the Model Repository Multimaster Additions now, enter `n`.

**12** If you entered `y` in the previous step, skip this step. If you entered `n` in the previous step, invoke the Opsware Installer with the `-r` option to specify the response file created by the latest interview. For example:

```
/opsware_system/opsware_installer/install_opsware.sh -r /
usr/tmp/oiresponse.stand_to_mm
```

**13** At the components prompt, select one or more components to install:

```
Welcome to the Opsware Installer.
Please select the components to install.
1 ( ) Model Repository (truth), Multimaster Additions
2 ( ) Data Access Engine (spin), Multimaster Component
```

```
3 ( ) Multimaster Infrastructure Components (vault)
4 ( ) Command Engine (way), Multimaster Component
5 ( ) Software Repository (word), Multimaster Component
6 ( ) Opsware Global Filesystem, Multimaster Component
7 ( ) Opsware Command Center (OCC), Multimaster Component
Enter a component number to toggle ('a' for all, 'n' for
none).
When ready, press 'c' to continue, or 'q' to quit.

Selection:
```

You must install the components in the order they are listed. For example, you must install the Model Repository Multimaster Additions first.

If you are installing all of the components on a single server, then you may enter `a` for all. If you do not select `a`, then you must run the Opsware Installer again (as shown in the preceding step) and select the remaining components.

**14** If you are installing the components on multiple servers, follow the instructions in this step. (If you are installing the components on a single server, skip this step.)

Copy the response file generated by the installer interview to all other servers in the source core.

On each server in the source core, run the Opsware Installer with the `-r` option, as shown in step 12. Select and install the remaining components from the menu shown in step 13.

You must install each multimaster addition on the same server running the corresponding standalone component. For example, install the Model Repository Multimaster Additions on the server running the standalone Model Repository, and install the Data Access Engine Multimaster Component on the server running the standalone Data Access Engine. Although not required, the Model Repository Multimaster Component (vault) is usually installed on the same server as the Model Repository.

**15** Follow the instructions in the section “Adding a Core to a Multimaster Mesh” on page 98.

## Adding a Core to a Multimaster Mesh

Before proceeding with the installation, follow the instructions in “Prerequisites for a Multimaster Installation” on page 93.

This section describes how to add a new Opsware core to a multimaster mesh. Throughout this section, the first core in the mesh is referred to as the source core. The new core that you are adding is called the target core. (If you do not have a multimaster mesh, you are reading the wrong section; go to the section “Converting a Core from Standalone to Multimaster” on page 94.)

Perform the following steps to add a new core to a multimaster mesh:

1. Obtain the Opsware SAS installation media for this release.

   See “Installation Media for the Opsware Installer” on page 62, including the recommendation, “Copying the DVD to a Local Disk.”

2. On the Model Repository server of the source core and on each server of the target core, mount the DVD or NFS-mount the directory that contains a copy of the DVD contents.

   The Opsware Installer must have read/write root access to the directories where it installs Opsware components, even NFS-mounted network appliances.

3. On the Model Repository server in the source core, invoke the Opsware Installer with the `-r` (response file) and the `--interview` options. For example:

   ```
   /opsware_system/opsware_installer/install_opsware.sh -r /
   usr/tmp/oiresponse.stand_to_mm --interview
   ```

   You must specify the response file created when you converted the core from standalone to multimaster.

   The Opsware Installer displays the following options:

   ```
   Welcome to the Opsware Installer. Please select one of the
   following installation options:

   1 - Standalone Installation: Standalone Opsware Core
   2 - Multimaster Installation: First Core (convert from
   standalone)
   3 - Multimaster Installation: Define New Facility; Export
   Model Repository
   ```

```
4 - Multimaster Installation: Additional Core
5 - Satellite Installation: Opsware Satellite
```

4 At the installation options prompt, select the following option:

```
3 - Multimaster Installation: Define New Facility; Export
Model Repository
```

5 At the interview mode prompt, select one of the following options:

```
1 - Simple Interview Mode
2 - Advanced Interview Mode
```

Option 1 is for using default values for many of the configuration parameters. Option 2 is for specifying all configuration parameters during the interview.

6 Respond to the interview prompts.

The installer displays default values in square brackets [ ].

For the short name of the target core (`slaveTruth.dcNm` parameter), enter a new facility name. This name must be unique within the multimaster mesh.

See "Required Information for Running the Installer Interview" on page 41.

7 Decide if you want to finish the interview.

When you enter all of the required information, the Opsware Installer displays this message:

```
All parameters have values. Do you wish to finish the
interview (y/n):
```

If you are satisfied with your answers, press y.

If you want to review or change your answers, press n. The installer displays the prompts again, showing in brackets [ ] the values that you previously entered.

8 Create the response file.

When you are finished with the interview, the installer prompts you for the name of the response file:

```
Name of response file to write
[/usr/tmp/oiresponse.add_dc_to_mesh]
```

The response file is a text file that contains the answers you entered during the interview. You can enter the name of the response file or accept the default. In either case, write down the name of the response file. Note that the default file name corresponds to the type of installation.

**9** The Opsware Installer prompts you to indicate whether you want to continue the installation by using the response file. Select one of the following options:

- If the Opsware Gateway in the source core is on a different server than the Model Repository, enter `n`. Copy the response file to the server with the Opsware Gateway and go on to the next step.
- If you are satisfied with the responses you entered in the interview and you are ready to define the new facility now, enter `y` to continue.
- If you do not want to define the new facility now, enter `n`.

**10** If you entered `y` in the previous step, skip this step. If you entered `n` in the previous step, log into the server running the Opsware Gateway and invoke the installer with the `-r` option. Be sure to specify the response file created by the latest interview. For example:

```
/opsware_system/opsware_installer/install_opsware.sh -r /
usr/tmp/oiresponse.add_dc_to_mesh
```

**11** At the components prompt, select the following option:

```
1 ( ) Define New Facility
```

Wait for the installer to finish this operation before going on to the next step. The Opsware Installer enters the target facility in the Model Repository of the source core, automatically generating the target facility's ID.

**12** Find the ID of the target facility.

To find the facility ID, perform the following steps:

- Log into the Opsware Command Center as the `admin` user at the source facility.
- From the navigation panel, select Facilities under Environment.
- Click the link for the target facility. Write down the facility ID.

In step 13 through step 19, you perform the tasks for exporting data from the Model Repository of the source core.

If you are adding a third (or more) core to a multimaster mesh, you can export data from a core other than the original source core. In this case, the instructions are slightly different, as noted in step 14 on page 101 and step 35 on page 106.

**13** On the servers where the Data Access Engine (spin) is installed, stop the engine by entering the following command:

```
/etc/init.d/spin stop
```

If the Opsware Command Center and the Data Access Engine are installed on different servers, you must also run the preceding command on the Opsware Command Center server.

**14** On the server running the Model Repository Multimaster Component, wait for all transactions to be published by examining the `/var/lc/vault/log` file.

If the log contains successive entries "QUERIED THE DATABASE" and does not contain recent "SENDING TRANSACTION" entries, the transactions from the installation have been published.

If you are going to export data from a core other than the original source core, wait for the transactions to propagate to the core that will be exported before performing step 17 on page 101.

**15** On the server where the Model Repository Multimaster Component (vault) is installed, stop the engine by entering the following command:

```
/etc/init.d/vaultdaemon stop
```

**16** Log into the server running the Model Repository and invoke the installer with the `-r` option to specify the response file created by the latest interview. For example:

```
/opsware_system/opsware_installer/install_opsware.sh -r /
usr/tmp/oiresponse.add_dc_to_mesh
```

**17** At the components prompt, select the following option:

```
2 ( ) Export Model Repository (truth)
```

The installer exports the data from the Model Repository into the `truth_data.tar.gz` file, which by default resides in the directory `/var/lc/truth`. (You specified this directory at the `truth.dest` prompt of the interview.)

Depending on the amount of data, the export might take 20 minutes or more. To track the progress of the export in a different window, run the following command.

```
tail -f /var/lc/install_opsware/truth/truth_exp<number>.log
```

**18** On the source core servers where the Data Access Engine (spin) is installed, start the engine by entering the following command:

```
/etc/init.d/spin start
```

If the Opsware Command Center and the Data Access Engine are installed on different servers, you must also run the preceding command on the Opsware Command Center server.

19 On the server where the Model Repository Multimaster Component (vault) is installed, start the engine by entering the following command:

```
/etc/init.d/vaultdaemon start
```

Examine the logs for the Model Repository Multimaster Component to ensure that it started properly. These logs are located in the following directory:

```
/var/lc/vault
```

The log files are named `log`, `log.1`, `log.2`, `log.3`, and so forth.

20 Copy the Model Repository export file (`truth_data.tar.gz`) to the server where you will install the Model Repository in the target core.

The Unix `oracle` user needs read access to the `truth_data.tar.gz` file on the Model Repository host in the target core.

21 Copy the global response file (`oiresponse.global`) from the source core server of the Model Repository to the target core server on which you will install the new Model Repository.

On the source core, the `oiresponse.global` file resides in the same directory as the Model Repository export file. The default directory is `/var/lc/truth`.

22 On the target core servers, make the following directory:

```
mkdir -p /var/lc/crypto/cadb/realm
```

23 Copy the database of cryptographic material from the source core server that is running the Model Repository to every target core server. The database of cryptographic material is in the following file:

```
/var/lc/crypto/cadb/realm/opsware-crypto.db.e
```

The full path name of the file on the target core servers must match the preceding line. The root user requires read access to the directory and file.

24 Log into the target core server on which you will install the Model Repository and invoke the Opsware Installer. Specify the `-r oiresponse.global` file and the `--interview` options. For example:

```
/opsware_system/opsware_installer/install_opsware.sh -r /
usr/tmp/oiresponse.global --interview
```

Be sure to specify the global response file that you copied to the target core.

The Opsware Installer displays following options:

```
Welcome to the Opsware Installer. Please select one of the
following installation options:

1 - Standalone Installation: Standalone Opsware Core
2 - Multimaster Installation: First Core (convert from
standalone)
3 - Multimaster Installation: Define New Facility; Export
Model Repository
4 - Multimaster Installation: Additional Core
5 - Satellite Installation: Opsware Satellite
```

**25** At the installation options prompt, select the following option:

```
4 - Multimaster Installation: Additional Core
```

**26** At the interview mode prompt, select one of the following options:

```
1 - Simple Interview Mode
2 - Advanced Interview Mode
```

Option 1 is for using default values for many of the configuration parameters. Option 2 is for specifying all configuration parameters during the interview.

**27** Respond to the interview prompts.

The installer displays default values in square brackets [ ]. Unless you have changed the source core, do not change the values that were in the global response file you copied from the source core. Note the following requirements for the prompts:

- The facility ID, short name, and subdomain must match the values generated when the target facility was defined in the source core. You wrote down the facility ID in step 12 on page 100.
- The authorization domain must match the value provided for the source core.
- The path to the data export file, `truth_data.tar.gz`, in the target core must match the path you used when copying the file from the source core.
- The path for the OS provisioning media must already exist on the server where you will install the OS Provisioning Media Server component.

**28** Decide if you want to finish the interview.

When you enter all of the required information, the Opsware Installer displays this message:

```
All parameters have values. Do you wish to finish the
interview (y/n):
```

If you are satisfied with your answers, press y.

If you want to review or change your answers, press n. The installer displays the prompts again, showing in brackets [ ] the values that you previously entered.

**29** Create the response file.

When you are finished with the interview, the installer prompts you for the name of the response file:

```
Name of response file to write
[/usr/tmp/oiresponse.mm_subs]
```

The response file is a text file that contains the answers you entered during the interview. You can enter the name of the response file or accept the default. In either case, write down the name of the response file. Note that the default file name corresponds to the type of installation.

**30** The Opsware Installer prompts you to indicate whether you want to continue the installation by using the response file. Select one of the following options:

- If you are satisfied with the responses you entered in the interview and you are ready to install the Model Repository now, enter `y` to continue.
- If you do not want to install the Model Repository now, enter `n`.

**31** If you entered `y` in the previous step, skip this step. If you entered `n` in the previous step, invoke the Opsware Installer with the `-r` option to specify the response file created by the interview. For example:

```
/opsware_system/opsware_installer/install_opsware.sh -r /
usr/tmp/oiresponse.mm_subs
```

**32** At the components prompt, select one or more components to install:

```
Welcome to the Opsware Installer.
Please select the components to install.
1 ( ) Model Repository, Consumer Multimaster Component (truth
slave)
2 ( ) Data Access Engine (spin), Multimaster Component
3 ( ) Multimaster Infrastructure Components (vault)
4 ( ) Command Engine (way), Multimaster Component
5 ( ) Software Repository (word), Multimaster Component
6 ( ) Opsware Global Filesystem, Multimaster Component
7 ( ) Opsware Global Filesystem Server (OGFS)
8 ( ) Opsware Documentation
9 ( ) Opsware Command Center (OCC), Multimaster Component
10 ( ) OS Provisioning Media Server
11 ( ) OS Provisioning Build Manager
```

```
12 ( ) Gateway - HA Configuration, Slave Core
13 ( ) OS Provisioning Boot Server
Enter a component number to toggle ('a' for all, 'n' for
none).
When ready, press 'c' to continue, or 'q' to quit.

Selection:
```

You must install the components in the order they are listed. For example, you must install the Model Repository first.

If you are installing all of the components on a single server, then you may enter `a` for all. If you do not select `a`, then you must run the Opsware Installer again (as shown in the preceding step) and select the remaining components. (If you are installing the components on multiple servers, see the next step.)

**33** If you are installing the components on multiple servers, follow the instructions in this step. (If you are installing the components on a single server, skip this step.)

Copy the response file generated by the installer interview to all other servers in this core.

Copy the `tnsnames.ora` file from the server with the Model Repository to the other core servers. The directory path for the file must be the same on all core servers. (By default, `tnsnames.ora` is in the `/var/opt/oracle` directory.)

On each server in this core, run the Opsware Installer with the `-r` option, as shown in step 31. Select and install the remaining components from the menu shown in step 32.

You must install the Opsware Documentation component on the server where you install the Opsware Command Center component.

If the Model Repository exists on a server with no other Opsware components installed on it, you must install an Opsware Agent on that server. See the *Opsware® SAS User's Guide* for instructions.

**34** (Optional) If you are distributing the core components across multiple servers, you may install additional instances of the following components:

- Data Access Engine

  If you install more than one Data Access Engine, then you must perform the procedure described in "Reassigning the Data Access Engine to a Secondary Role" in the *Opsware® SAS Administration Guide*.

- OS Provisioning Media Server
- Opsware Command Center
- Opsware Global File System

35 If you exported data from a core other than the original source core, you might need to configure TIBCO manually.

By default, the target core will try to connect to the original source core. If you want the target core to connect to a different core then you must configure TIBCO manually and edit the Opsware Gateway properties file. For instructions, see "Adding a TIBCO Rendezvous Neighbor" on page 153.

36 Perform the tasks in Chapter 5, "Post-Installation Tasks" on page 73 of this guide.

37 Perform the tasks in the following section.

## Multimaster Post-Installation Tasks

After you add a new core to a multimaster mesh, perform the tasks described in this section.

### Associating Customers with a New Facility

Associate the appropriate customers with each new facility so that servers managed at that facility are associated with the correct customers accounts. For more information, see the Customer Account Administration section of the *Opsware® SAS Configuration Guide*.

### Updating Permissions for New Facilities

After you add new facilities to your multimaster mesh, your Opsware users will not have the required permissions to access these new facilities. To grant access, you must assign the required permissions to the user groups. For more information, see the User Group and Setup section of the *Opsware® SAS Configuration Guide*.

### Verifying Multimaster Transaction Traffic

Perform the following steps to verify multimaster transaction traffic with the target facility:

1 Log into the Opsware Command Center as a user that belongs to the Opsware System Administrators group.

2. From the navigation panel, click Multimaster Tools under Administration. The State View window appears.

3. In the State View Window, note the color of the status box beside each transaction.

   A transaction is a unit of change to a Model Repository database that consists of one or more updates to rows and has a globally unique transaction ID. If the transactions with the target facility are green, the new Opsware core is integrated into the multimaster mesh. It is normal for some of the transactions to have an orange status (not sent) for a while.

4. Click **Refresh** to refresh the cached data.

For more information, see the Opsware Multimaster Mesh Administration section in the *Opsware® SAS Administration Guide*.

# Chapter 7: Opsware Satellite Installation

**IN THIS CHAPTER**

This section discusses the following topics:

- Overview of Satellite Installation
- Satellite Requirements
- Gateway Configuration for a Satellite
- Satellite Installation
- Post-Installation Tasks for a Satellite

## Overview of Satellite Installation

An Opsware Satellite manages servers in a remote data center. The following steps provide an overview of the Satellite installation process. For detailed instructions, see "Satellite Installation" on page 120.

1. Obtain the Opsware SAS installation DVD.
2. Run the Opsware Installer (`install_opsware.sh` script) in interview mode. The interviewer prompts you for information about your environment and saves the information in a response file.
3. Run the Opsware Installer and select the Opsware Gateway from the list of components to install. The Opsware Installer launches the Opsware Gateway Installer.
4. Respond to the prompts of the Opsware Gateway Installer.
5. Run the Opsware Installer (`install_opsware.sh` script) and select the other components to install.

## Satellite Requirements

Before you install an Opsware Satellite, verify that the requirements detailed in the following sections are met.

### Open Ports Required for a Satellite

The ports listed in Table 7-1 must be open for the Opsware Gateway in a Satellite. The ports in the table are the default values. (You may select other values during the installation.)

*Table 7-1: Open Ports for a Satellite Gateway*

| PORT | PROPERTY NAME IN OPSWARE GATEWAY PROPERTIES FILE | DESCRIPTION |
|---|---|---|
| 2001 | `opswgw.TunnelDst` | The port for a tunnel end-point listener. This port will be used if you install other Gateways that tunnel to the Gateway on this Satellite. |
| 3001 | `opswgw.ProxyPort` | The proxy port on which the Agents contact the Gateway. |
| 4040 | `opswgw.IdentPort` | The port of the Gateway's `ident` service, which is used by the Software Repository Cache. |

If you are going to install the OS Provisioning Boot Server and Media Server in the Satellite, then additional ports must be open. For a list of these ports, see Table 2-2 on page 33.

### Entries Required in /etc/hosts for a Satellite

The Software Repository Cache of the Satellite requires the following entries in the `/etc/hosts` file:

```
127.0.0.1  theword
127.0.0.1  wordcache
```

### Required Packages for SuSE Linux Enterprise Server 9

For a Satellite running this version of Linux, the following packages must be installed:

- `nfs-utils` (for OS Provisioning Boot Server)

- `xinetd` (for `tftp`)
- `sharutils` (for `uuencode` and `uudecode` in ADT)
- `compat-2004.7.1-1.2` (for ADT, `compat` includes `libstdc++`)

### Other Requirements for a Satellite

The following requirements must also be met:

- The Satellite server meets the requirements listed in the section, "Supported Operating Systems," of the guide, *Planning Deployments for Opsware® SAS*. The supported operating systems for the OS Provisioning components are not the same as those for the other Satellite components (Gateway and Software Repository Cache).
- The Satellite server must have the necessary packages listed in "Operating System Requirements" on page 29.
- The Opsware core for this Satellite is up and running.
- The Satellite server must have network connectivity to the server running the core Gateway.
- In the Opsware Command Center for the core, you can log in as a member of the Administrators group (`admin`) and as a member of a group that has the Manage Gateway permission.
- You have root access on the core server so that you can copy the database of cryptographic material from the core to the Satellite server.
- The Satellite server uses UTC, as described in "Time and Locale Requirements" on page 40. The time of the Satellite server must be synchronized with the core server.
- When using network storage for the Software Repository Cache, the network storage configuration must allow root write access over NFS to the directories where the Software Repository Cache is to be installed.
- If you are going to install the OS Provisioning Boot Server and Media Server in the Satellite, then see the requirements in "DHCP Proxying" on page 35.
- You know how to edit files with the `vi` editor. The Opsware Gateway Installer launches the `vi` editor, which you will use to edit a properties file.

## Gateway Configuration for a Satellite

This section illustrates various Satellite topologies and the corresponding settings in the Gateway properties files. In the diagrams, the arrows between Gateways represent tunnels. (A tunnel is a TCP connection between two Gateways that carries multiplexed TCP or UDP connections.) The boxes labelled with the letter “A” designate managed servers, which run Opsware Agents.

### Satellite with a Standalone Core

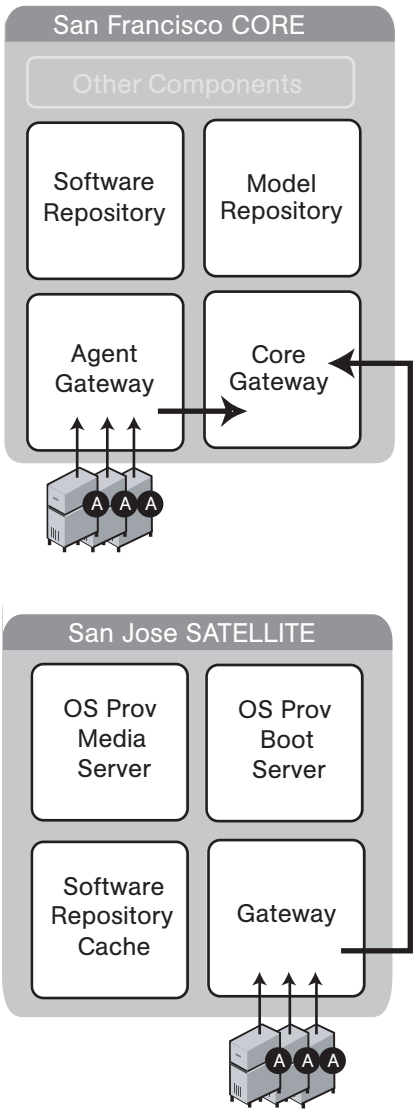
Figure 7-1 shows a single Opsware Satellite that has a tunnel to a standalone core. In this example, the main facility is in San Francisco, and a smaller remote facility is in San Jose.

The core is made up of several components, including the Software Repository, the Model Repository, and two gateways. The figure does not show other required core components, such as the Command Engine, but indicates them with an ellipsis (...) button. When you install a standalone core, the Opsware Installer creates both the Agent and core Gateways. The Agents in the San Francisco facility communicate with the core through the Agent Gateway. The Agents in the San Jose facility connect to the San Francisco core via TCP connections to the Satellite Gateway.

In a Satellite, the Software Repository Cache and Gateway components are required. The Software Repository Cache contains local copies of software packages to be installed on managed servers in the Satellite. The Gateway multiplexes connections into and out of the Satellite via one or more tunnels. Optionally, a Satellite can contain the OS Provisioning Boot Server and Media Server components.

*Figure 7-1: Single Satellite With a Standalone Core*




The following listing shows a few entries in the Gateway properties file of the San Jose Satellite.

In the properties file, the `opswgw.GWAddress` specifies the IP address or host name where the Satellite Gateway runs. When a new Gateway is added to a realm, the value of the `opswgw.GWAddress` is dynamically added to the list of Gateways that Agents in the

realm can communicate with. (A realm is a routable group of IP addresses.) The Agent installer and the `opswgw.GWAddress` must both specify either IP addresses or host names. For example, if the Agent installer specifies an IP address in its `opsw_gw_addr_list` option, then the `opswgw.GWAddress` must also specify an IP address, not a host name. If host names are used, they must be resolvable (with DNS or `/etc/hosts`) by the Agents that contact this Gateway. Specifying IP addresses is recommended because it is less error prone. (This document shows host names in the example diagrams and listings because they are easier to read.)

The `opswgw.Realm` specifies the realm of the Gateway. A realm is a logical name for a group of IP addresses that can be contacted by a particular set of Gateways. Realms enable Opsware SAS to manage servers with overlapping IP addresses. (This situation can occur when the servers in a remote facility are behind NAT devices or firewalls.) The realm plus the IP address uniquely identifies a managed server. Servers with overlapping IP addresses must reside in separate realms.

The `opswgw.TunnelSrc` has five parameters. The first two parameters identify the remote host (`sanfran.myops.com`) and port (2001) where the core Gateway listens for connections. Note that the host and port of the `opswgw.TunnelSrc` in the Satellite must match those of the `opswgw.TunnelDst` in the core. The next two parameters of `opswgw.TunnelSrc` specify the cost and bandwidth of the tunnel. (See “Configuring Routing (Cost)” on page 116 and “Limiting Bandwidth” on page 120.) The last parameter (`.../opswgw.pem`) is a certificate file in the Privacy Enhanced Mail (PEM) format. If you specify the certificate file, the data transmitted through the tunnel will be encrypted using SSL. The header of the certificate file includes the cipher choice and authentication options.

The `opswgw.DoNotRouteService` and `opswgw.HijackService` properties are required for this Satellite Gateway because the Satellite includes a Software Repository Cache. With these properties, if an Agent has a request for the Software Repository, the Satellite Gateway routes the request to the local Software Repository Cache.

The `opswgw.ProxyPort` identifies the port on the Satellite through which the Agents contact the Gateway. The `opswgw.IdentPort` is for an identity service used by the Software Repository Cache.

Typically, you'll use the default ports for the properties. However, you must enter the hosts for the `opswgw.GWAddress` and `opswgw.TunnelSrc` properties. The following listing shows some of the entries in the Gateway properties file for the San Jose Satellite. (Although the `opswgw.TunnelSrc` entry wraps around to the next line in this listing, in the actual properties file the entry is on a single line.)

```
opswgw.Gateway=SanJose
opswgw.Realm=SanJose
opswgw.GWAddress=sanjose.myops.com
opswgw.TunnelSrc=sanfran.myops.com:2001:10:0:/var/opt/OPSWgw/
SanJose/opswgw.pem
opswgw.DoNotRouteService=theword:1003
opswgw.DoNotRouteService=127.0.0.1:1003
opswgw.HijackService=wordcache:1003
opswgw.ProxyPort=3001
opswgw.IdentPort=4040
```

The following lines are from the core Gateway properties file of the San Francisco facility:

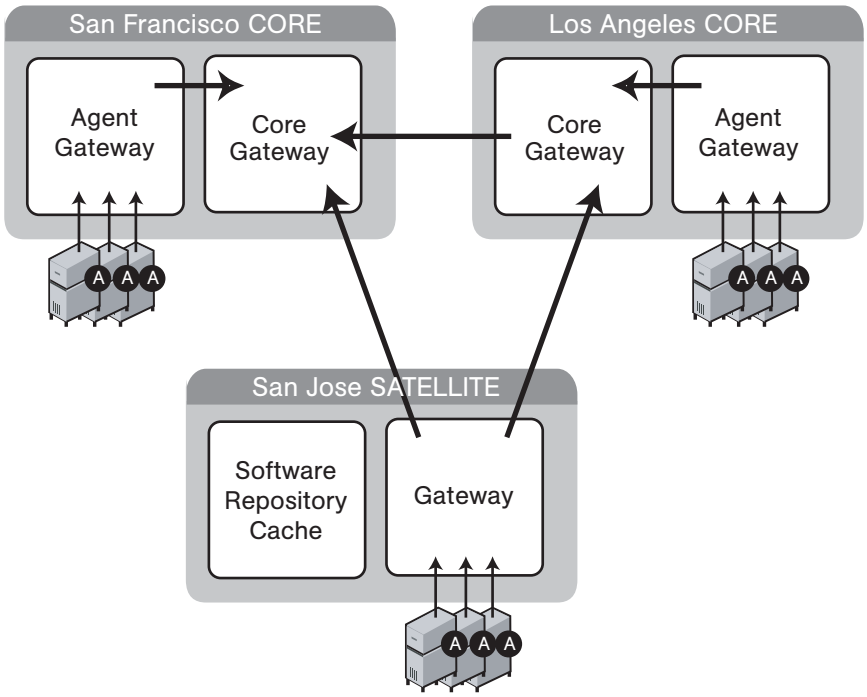
```
opswgw.Gateway=cgw0-SanFrancisco
opswgw.Realm=SanFrancisco
opswgw.TunnelDst=2001:/var/opt/OPSWgw/cgw0-SanFrancisco/
opswgw.pem
```

### Satellite in a Mulitmaster Mesh

Figure 7-2 shows two cores, San Francisco and Los Angeles, in a multimaster mesh. The multimaster traffic passes through the core Gateways. The Gateway in the San Jose Satellite points to both core Gateways. In this example, the communication link between the San Jose and San Francisco facilities is the fastest and has the most bandwidth. During normal operations, the servers in San Jose are managed by the San Francisco

core. If the connection between San Jose and San Francisco fails, then the Gateway in San Jose will communicate instead with the core in Los Angeles. (See “Configuring Routing (Cost)” on page 116.)

*Figure 7-2: Single Satellite in a MultiMaster Mesh*




The lines that follow are from the properties file of the Satellite Gateway in San Jose. The first `opswgw.TunnelSrc` property points to the San Francisco Gateway; the second one points to the Los Angeles Gateway. Both lines indicate that the core Gateways use the default port (2001) to listen for connections.

```
opswgw.Gateway=SanJose
opswgw.Realm=SanJose
opswgw.TunnelSrc=sanfran.myops.com:2001:100:0:/var/opt/OPSWgw/
SanJose/opswgw.pem
opswgw.TunnelSrc=losang.myops.com:2001:200:0:/var/opt/OPSWgw/
SanJose/opswgw.pem
```

### *Configuring Routing (Cost)*

A Satellite Gateway routes traffic to only one core Gateway at any given time. The Gateway chooses the route with the lowest cost. The cost is the third parameter of the `opswgw.TunnelSrc` property. In the preceding listing, the `opswgw.TunnelSrc`

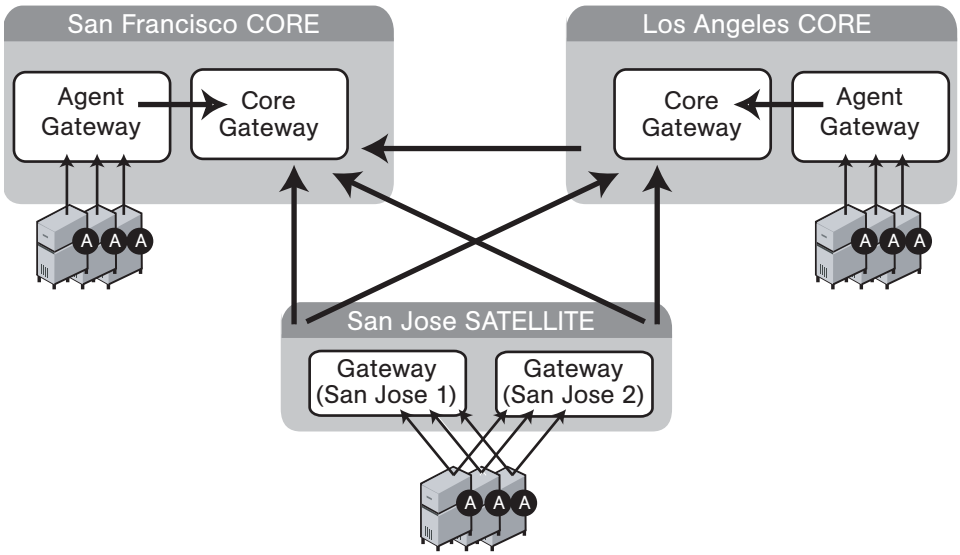
 

properties specify that the cost from San Jose to San Francisco is 100 and the cost between San Jose and Los Angeles is 200. Therefore, the Satellite Gateway will use the connection to San Francisco, unless for some reason that connection becomes unavailable.

### Multiple Gateways in a Satellite

The topology shown in Figure 7-3 provides failover capability in two ways. First, each Gateway in the San Jose Satellite tunnels to both core Gateways. If one core becomes unavailable, the other core can manage the servers in the Satellite. Second, the Agents in the San Jose Satellite point to both Satellite Gateways. If one Satellite Gateway becomes unavailable, the Agents on the managed servers can communicate with a core Gateway via the other Satellite Gateway. Both Gateways in San Jose must belong to the same realm. An Agent can communicate with any Gateway in the same realm.

*Figure 7-3: Multiple Gateways in a Satellite*




The following lines are from the core Gateway properties file of the San Francisco facility:

```
opswgw.Gateway=cgw0-SanFrancisco
opswgw.Realm=SanFrancisco
opswgw.TunnelDst=2001:/var/opt/OPSWgw/cgw0-SanFrancisco/
opswgw.pem
```

The core Gateway properties file of the Los Angeles facility has similar entries:

```
opswgw.Gateway=cgw0-LosAngeles
opswgw.Realm=LosAngeles
```

```
opswgw.TunnelDst=2001:/var/opt/OPSWgw/cgw0-LosAngeles/
opswgw.pem
opswgw.TunnelSrc=sanfran.myops.com:2001:1:0:/var/opt/OPSWgw/
cgw0-LosAngeles/opswgw.pem
```

The lines that follow are from the properties file of the first Gateway in the San Jose Satellite:

```
opswgw.Gateway=SanJose1
opswgw.Realm=SanJose
opswgw.TunnelSrc=sanfran.myops.com:2001:100:0:/var/opt/OPSWgw/
SanJose1/opswgw.pem
opswgw.TunnelSrc=losang.myops.com:2001:200:0:/var/opt/OPSWgw/
SanJose1/opswgw.pem
```

The next lines are from the properties file of the second Gateway in the San Jose Satellite:

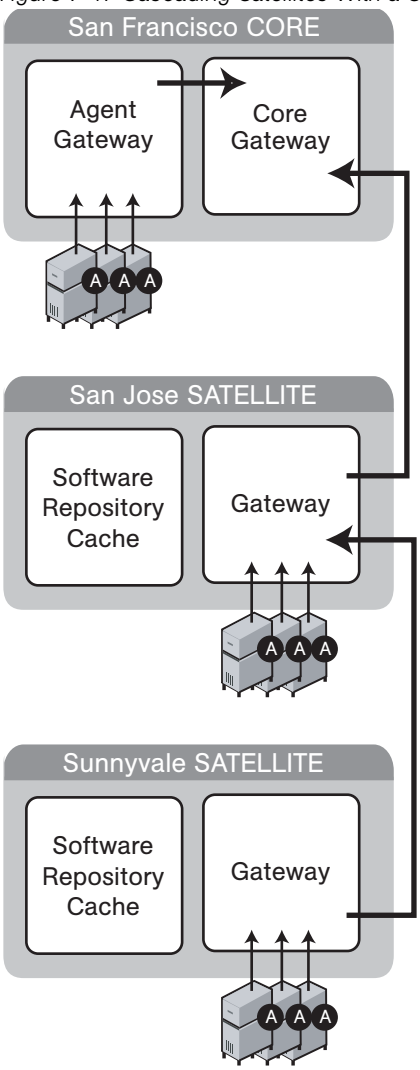
```
opswgw.Gateway=SanJose2
opswgw.Realm=SanJose
opswgw.TunnelSrc=sanfran.myops.com:2001:100:0:/var/opt/OPSWgw/
SanJose2/opswgw.pem
opswgw.TunnelSrc=losang.myops.com:2001:200:0:/var/opt/OPSWgw/
SanJose2/opswgw.pem
```

## Cascading Satellites

Figure 7-4 is an example of cascading Satellites, a topology in which Satellite Gateways are connected in a chain. These Satellite Gateways must be in different realms. (For more information, see "Managing the Software Repository Cache" in the *Opsware® SAS Administration Guide*.)

*Figure 7-4: Cascading Satellites With a Standalone Core*




The following lines are from the core Gateway properties file of the San Francisco facility:

```
opswgw.Gateway=cgw0-SanFrancisco
opswgw.Realm=SanFrancisco
```

```
opswgw.TunnelDst=2001:/var/opt/OPSWgw/cgw0-SanFrancisco/
opswgw.pem
```

The lines that follow are from the Gateway properties file of the San Jose Satellite.

```
opswgw.Gateway=SanJose
opswgw.Realm=SanJose
opswgw.TunnelDst=2001:/var/opt/OPSWgw/SanJose/opswgw.pem
opswgw.TunnelSrc=sanfran.myops.com:2001:100:0:/var/opt/OPSWgw/
SanJose/opswgw.pem
```

The next lines are from the Gateway properties file of the Sunnyvale Satellite:

```
opswgw.Gateway=Sunnyvale
opswgw.Realm=Sunnyvale
opswgw.TunnelSrc=sanjose.myops.com:2001:100:256:/var/opt/
OPSWgw/Sunnyvale/opswgw.pem
```

#### *Limiting Bandwidth*

In Figure 7-4, suppose that the tunnel between Sunnyvale and San Jose shares a 512 kilobit/sec DSL connection with another application. Since this connection is relatively slow, you might want to limit the tunnel bandwidth to 256 kilobits/sec. To limit the bandwidth, you specify 256 for the fourth parameter of the `opswgw.TunnelSrc` property. (See the previous listing of the Sunnyvale properties file.) If you do not want to limit the tunnel bandwidth, set this parameter to 0. Note that the bandwidth parameter is not used to determine the cost of a route. (See “Configuring Routing (Cost)” on page 116.)

## Satellite Installation

This section describes how to create a new Opsware Satellite with the simple topology shown in Figure 7-1. This topology has the following characteristics:

- The Satellite contains one Opsware Gateway and one Software Repository Cache, installed on the same server.
- The Satellite Gateway communicates with one core Gateway. No other Gateways communicate with the Satellite Gateway.

### Required Information for Installing a Satellite

You will be prompted for the following information during the installation process:

- The password to decrypt cryptographic material. During the installation of the core, the Opsware Installer prompts for this password.

- The IP address of the server running the core Gateway.
- The IP address of the server on which you will install the Satellite Gateway.
- The port of the tunnel destination of the core Gateway. (The default port is 2001.) The core Gateway listens on this port for a connection from the Satellite Gateway. In the core Gateway properties file, this port is the value of the `opswgw.TunnelDst` property. On the core Gateway server, the path of the properties file is as follows:

  `/var/opt/OPSWgw/cgw0-<facility>/opswgw.properties`
- The Opsware user name (`admin`) and password of a user that belongs to the Administrators group.
- The name of the new Gateway in the Satellite. The new Gateway will be installed in the following directory:

  `/var/opt/OPSWgw/<gateway-name>`
- The name of the new realm to be serviced by the Gateway in the Satellite. Opsware SAS uses the realm name and the IP address of a managed server to uniquely identify a managed server. The Opsware Gateway Installer assigns the realm name to the new facility name of the Satellite. The core and Satellite facility names will be different. You may want to name the realm according to the physical location of the Satellite's data center, for example, the building, corporate site, or city. The Opsware Command Center lists the facility names of the core and its Satellites.

### Installing a Satellite

This section contains the step-by-step instructions for running the Opsware Installer (`install_opsware.sh` script).

1. Obtain the Opsware Server Automation System (SAS) installation media.

   See "Installation Media for the Opsware Installer" on page 62, including the recommendation, "Copying the DVD to a Local Disk."

2. On the server where you will install the new Opsware Satellite, mount the DVD or NFS-mount the directory that contains a copy of the DVD contents.

   The Opsware Installer must have read/write root access to the directories where it installs Opsware components, even NFS-mounted network appliances.

3. In a terminal window log in as root.

4. Make the following directory:
   `mkdir -p /var/lc/crypto/cadb/realm`

5 Copy the database of cryptographic material from the core server to the Satellite server. On the core server, this database is in the following file:

```
/var/lc/crypto/cadb/realm/opsware-crypto.db.e
```

The database of cryptographic material must be copied to the same directory and file name on the Satellite server. The directory and database need to be readable by the root user.

6 Change to the root directory:

```
cd /
```

7 Run the Opsware Installer in interview mode by invoking it with no command-line options:

```
/opsware_system/opsware_installer/install_opsware.sh
```

You must specify the full path to the script. The directory path shown in this step indicates that you copied the Opsware SAS DVD to a local disk or a network share using the required directory structure.

The Opsware Installer displays the following options:

```
Welcome to the Opsware Installer. Please select one of the
following installation options:

1 - Standalone Installation: Standalone Opsware Core
2 - Multimaster Installation: First Core (convert from
standalone)
3 - Multimaster Installation: Define New Facility; Export
Model Repository
4 - Multimaster Installation: Additional Core
5 - Satellite Installation: Opsware Satellite
```

8 At the installation options prompt, select the following option:

```
5 - Satellite Installation: Opsware Satellite
```

9 At the interview mode prompt, select one of the following options:

```
1 - Simple Interveiw Mode
2 - Advanced Interview Mode
```

Option 1 is for using default values for many of the configuration parameters. Option 2 is for specifying all configuration parameters during the interview.

10 Respond to the interview prompts.

The `cgw_address prompt` is for the core Gateway, not the Satellite Gateway. For more information on the prompts, see Table 3-6 on page 58.

11 Decide if you want to finish the interview.

When you enter all of the required information, the Opsware Installer displays this message:

```
All parameters have values. Do you wish to finish the
interview (y/n):
```

If you are satisfied with your answers, press y.

If you want to review or change your answers, press n. The installer displays the prompts again, showing in brackets [ ] the values that you previously entered.

**12** Create the response file.

When you are finished with the interview, the installer prompts you for the name of the response file:

```
Name of response file to write
[/usr/tmp/oiresponse.satellite]
```

The response file is a text file that contains the answers you entered during the interview. You can enter the full path and name of the response file or accept the default. Note that the default file name corresponds to the type of installation.

**13** The Opsware Installer prompts you to indicate whether you want to continue the installation by using the response file:

```
Would you like to continue the installation using this
response file? (y/n):
```

If you are satisfied with the responses you entered in the interview and you are ready to install the Satellite now, enter `y` to continue. If you do not want to install the Satellite now, enter `n`.

**14** If you entered `y` in the previous step, skip this step. If you entered `n` in the previous step, invoke the Opsware Installer with the `-r` option to specify the response file created by the interview:

```
/opsware_system/opsware_installer/install_opsware.sh -r
<full_path_to_response_file>
```

**15** At the components prompt, select 1 to install the Opsware Gateway. The components prompt follows:

```
Welcome to the Opsware Installer.
Please select the components to install.
1 ( ) Opsware Gateway (Interactive Install)
2 ( ) Software Repository Cache (wordcache)
```

```
3 ( ) OS Provisioning Boot Server
4 ( ) OS Provisioning Media Server
Enter a component number to toggle ('a' for all, 'n' for
none).
When ready, press 'c' to continue, or 'q' to quit.

Selection: 1
```

Note that you must install the components in the order they are listed.

**16** Verify that the Opsware Installer launches the Opsware Gateway Installer, which displays the following banner:

```
************************************************************
*                                                          *
*                Opsware Gateway Installer                 *
*         Copyright (C) 2004-2005: Opsware Inc.            *
*                   support@opsware.com                    *
*                                                          *
************************************************************
```

**17** Verify that you have the necessary information for the Gateway, as described in "Required Information for Installing a Satellite" on page 120. The Opsware Gateway Installer displays the following message:

```
For a new install please have the following information
available before you begin:

1) Opsware administrator username and password.
2) The Realm name this Gateway will service.
3) If the Realm is new what type will it be.
4) The unique Gateway name for this Gateway.

Are you ready to proceed? [y/n]
```

**18** At the proceeding prompt, enter y. The Opsware Gateway Installer displays the following lines:

```
================================================
ISM install
================================================
. . .
```

**19** Enter the name of the realm for the Opsware Gateway you are installing. The prompt for the realm follows:

```
================================================
Create/Verify Realm
================================================
Enter the Gateway's Realm name:
You entered '<realm-name>', is this correct [y/n]
```

**20** There are three ways for the installer to contact the Opsware core. At the prompt for the option number, enter 3. The installer displays the following lines:

```
I must now contact an Opsware Core to continue the
intallation...
There are three ways this can be done:
 1) Via an existing Gateway's ProxyPort
 2) Via direct connections (no NATs)
 3) Via a temporary (local) Gateway
Enter option number: 3
```

**21** Enter the IP address of the server running the core Gateway at the following prompt:

```
Enter IP of a remote GW:
```

**22** Enter the tunnel destination port of the core Gateway at the following prompt. The default port is 2001. (For more information, see "Required Information for Installing a Satellite" on page 120.)

```
Enter TunnelDst port of the remote GW: 2001
```

**23** At the following prompt, enter y.

```
Is the tunnel listener at <ip-addr:port>
using SSL? [y/n] y
```

**24** Enter the user name (`admin`) and password of an Opsware user that belongs to the Administrators group. The user name and password prompts follow.

```
================================================
Connect to Opsware
================================================

Log into Opsware as an administrator

Enter username:admin
Enter password:
```

**25** Verify that the Opsware Gateway Installer displays the following lines:

```
================================================
Checking time synchronization
================================================

Gateway time looks good.
```

**26** At the prompt that follows, enter 1 to create a new Satellite.

```
================================================
Configure Realm
================================================
```

```
The realm '<realm-name>' does not exist.  You have two
options:
  1) Create a new Satellite DC named '<realm-name>'.
  2) Add a new Realm, '<realm-name>', to an existing DC.
  3) Exit.
Enter option number: 1
```

**27** At the following prompt, enter the name for the new Opsware Gateway that you are installing.

```
================================================
Gateway Configuration
================================================

Enter the Gateway's name:
```

**28** Verify that the Opsware Gateway Installer opens the properties file in the `vi` text editor. The following lines are at the top of the properties file:

```
################################################
#
# Opsware Gateway Properties file for a SAT Gateway
#
################################################
```

The full path name of the properties file follows:

`/var/opt/OPSWgw/<gateway-name>/opswgw.properties`

**29** Opsware Gateway Properties File: For the `opswgw.GWAddress` property, enter the IP address of the host on which you are installing this Gateway (that is, the host you are logged into now). Example:

`opswgw.GWAddress=192.168.198.92`

**30** Opsware Gateway Properties File: For the `opswgw.TunnelSrc` property, change the placeholder IP address of 10.0.0.11 to the IP address of the host running the core Gateway. The port following the IP address is the tunnel destination of the core Gateway. (The default port is 2001.) Example:

`opswgw.TunnelSrc=192.168.165.242:2001:100:0:/var/opt/OPSWgw/<gateway-name>/opswgw.pem`

**31** Opsware Gateway Properties File: Because you are going to install a Software Repository Cache (wordcache) in a later step, verify that the following lines in the Opsware Gateway Properties file are not commented out:

```
opswgw.DoNotRouteService=theword:1003
opswgw.DoNotRouteService=127.0.0.1:1003
opswgw.HijackService=wordcache:1003
```

**32** After you've finished editing the Opsware Gateway Properties in `vi`, save the file and exit `vi`.

**33** Respond to the prompts that ask if you'd like to proceed. The Opsware Gateway Installer performs several more tasks and displays the following messages:

```
Gateway Crypto Generation
. . .
Wordcache Crypto Generation
. . .
Starting Opsware Gateway
. . .
Verify Gateway Startup
```

When it's finished, the Opsware Gateway Installer displays the following line:

```
Opsware Gateway Installed!
```

**34** Invoke the Opsware Installer with the `-r` option to specify the response file created by the interview in step 12 on page 123:

```
/opsware_system/opsware_installer/install_opsware.sh -r
<full_path_to_response_file>
```

**35** At the components prompt, select one or more components to install:

```
Welcome to the Opsware Installer.
Please select the components to install.
1 ( ) Software Repository Cache (wordcache)
2 ( ) OS Provisioning Boot Server
3 ( ) OS Provisioning Media Server

Enter a component number to toggle ('a' for all, 'n' for
none).
When ready, press 'c' to continue, or 'q' to quit.

Selection:
```

You must install the components in the order they are listed. For example, you must install the Software Repository Cache before the OS Provisioning Boot Server.

The Software Repository Cache is required and must be installed on the same server as the Gateway.

The OS Provisioning Boot Server and Media Server are required only if you want to use the Opsware OS Provisioning feature in the Satellite. The OS Provisioning Boot Server and Media Server can reside on a different server than the Gateway and Software Repository Cache. (See step 36.)

If you are installing all of the components on the same server, then you may enter `a` for all. If you do not select `a`, then you must run the Opsware Installer again (specifying the response file) and select the remaining components.

**36** If you are installing the OS Provisioning components on a different server than the other Satellite components, follow the instructions in this step.

- Copy the database of cryptographic material from the server with the Satellite Gateway to the server that will run the OS Provisioning components. Here is the full path of the database of cryptographic material:

  ```
  /var/lc/crypto/cadb/realm/opsware-crypto.db.e
  ```

  The database of cryptographic material must be copied to the same directory. The directory and database need to be readable by the root user.

- Copy the response file generated by the installer interview to the server that will run the OS Provisioning components.

- On the server that will run the OS Provisioning components, run the Opsware Installer with the `-r` option, as shown in step 34. Select and install the remaining components from the menu shown in step 35.

## Post-Installation Tasks for a Satellite

After you install a Satellite, perform the tasks listed in the following sections. For more information, see the Opsware Satellite Administration section of the *Opsware® SAS Administration Guide*.

### Facility Permission Settings

The Opsware Gateway Installer assigns the realm name to the facility name of the Satellite. To access managed servers in the Satellite, an Opsware user must belong to a group that has the necessary permissions for the Satellite's facility. Until you set the facility permissions, Opsware users cannot view or modify the managed servers associated with the Satellite's facility. For example, you might set the permissions for the Satellite facility to Read & Write for the Advanced Users group, enabling members of this group to modify the servers managed by the Satellite.

For instructions, see "Setting the Facility Permissions of a User Group" in the *Opsware® SAS Configuration Guide*.

### Checking the Satellite Gateway

To verify that the core Gateway is communicating with the Satellite Gateway, perform the following steps:

1. Log into the Opsware Command Center as a member of a users group that has the Manage Gateway permission.
2. From the navigation panel, select Administration ➤ Gateway.
3. Verify that the upper left corner of the Manage Gateway page displays a link for the new Satellite Gateway.

   If the Manage Gateway page does not display the link for the Satellite, you might need to correct the properties file of the Satellite Gateway. The full path name of the properties file follows:
   `/var/opt/OPSWgw/<gateway-name>/opswgw.properties`

   If you modify the properties file, you must restart the Satellite Gateway:
   `/etc/init.d/opswgw-<gateway-name> restart`
4. Log into the Opsware Command Center as a member of a users group that has the the Read (or Read & Write) permission on the Satellite's facility.
5. From the navigation panel, select Servers ➤ Manage Servers.
6. Verify that the Manage Server page displays the host name of the Satellite server.

### Enabling the Display of Realm Information

By default, the Opsware Command Center does not display realm information, which is needed by users who manage Gateways and Software Repository Caches.

To enable access to the realm information, perform the following steps:

1. Log into the Opsware Command Center as a user that belongs the Administrators group and to a group that has the Configure Opsware permission.
2. From the navigation panel, click Administration ➤ System Configuration.
3. Select the Opsware Command Center link.
4. In the System Configuration page, for the name `owm.features.Realms.allow`, type the value `true`.
5. Click **Save**.

### DHCP Configuration for OS Provisioning

After you install the OS Provisioning Boot Server component, you must set up a DHCP server. For more information, see “DHCP Configuration for OS Provisioning” on page 75.

# Chapter 8: What's Next

**IN THIS CHAPTER**

This section discusses the following topic:

- Configuration for Opsware SAS

## Configuration for Opsware SAS

After you've completed the tasks in the preceding sections of this guide, the core components of Opsware SAS should be running and you should be able to log into the Opsware Command Center. Now, Opsware SAS is ready to be configured so that end users can start managing servers in the operational environment. The configuration tasks follow:

- **Configure e-mail alerts for Opsware SAS**.

  The Opsware managed servers, the multimaster mesh, and the Opsware Code Deployment and Rollback feature can be configured to send e-mail alerts. Your e-mail administrator should set up the Opsware core and managed servers as sendmail clients. In the Opsware Command Center, you should configure the e-mail alerts before you install Agents on the managed servers. For more information, see the *Opsware® SAS Configuration Guide*.

- **Set up Opsware groups and users**.

  To log on to the Opsware Command Center, you specify a user name and password. Each user belongs to a group, and each group has a set of permissions for specific Opsware features. When the user logs on to the Opsware Command Center, only those features permitted by the user's groups are displayed. Each group also has permissions to perform read and write operations on managed servers that are associated with customers or facilities. For more information, see the *Opsware® SAS Configuration Guide*.

- **Create Opsware customers**.

  When you ran the Opsware Installer for a standalone core, you specified a default customer. You may also create and assign new customers to the facility. For more information, see the *Opsware® SAS Configuration Guide*.

- **Define the Software Tree**.

  The Opsware Command Center displays the Software Tree, a structure that organizes software by category, operating system, and version. The leaf nodes in the Software Tree represent the software packages that you can install on managed servers. You may associate these nodes with customers.

  There are three ways to create entries in the Software Tree:

  – Create the tree with the Opsware Command Center. For more information, see the *Opsware® SAS Configuration Guide*.

  – Import data with the DCML Exchange Tool (DET). The data you import is in a file created by exporting data from another Opsware core with the DET. For more information, see the *Opsware® SAS DCML Exchange Tool (DET) Reference Guide*.

  – Upload an Intelligent Software Module (ISM) into the core with the `ismtool`. An ISM contains the application bits (and optional control scripts) of a software package. After you upload the ISM, the package appears in the Software Tree of the Opsware Command Center and may be installed on managed servers. For more information, see *Opsware® SAS ISM Development Kit Guide*.

- **Install Opsware Agents on existing servers**.

  After you install an Opsware Agent, the server may be managed with Opsware SAS. For more information, see the *Opsware® SAS User's Guide*.

- **Prepare Opsware SAS for OS Provisioning**.

  When you provision (install) an OS on a server, Opsware SAS automatically installs an Agent. For more information on the setup tasks, see the *Opsware® SAS Configuration Guide*.

- **Prepare Opsware SAS for patch management**.

  For more information, see the *Opsware® SAS Configuration Guide*.

- **Prepare Opsware SAS for application provisioning**.

  For more information, see the *Opsware® SAS Configuration Guide*.

- **Establish monitoring practices for Opsware SAS by performing the following tasks**:
  - Monitor the log files of the Oracle database that underlies the Opsware Model Repository. For more information, see "Database Monitoring for the Model Repository" on page 147.
  - Run the Agent reachability tests in the Opsware Command Center. For more information, see the *Opsware® SAS User's Guide*.
  - Run the diagnostic tests in the Opsware Command Center. For more information, see the *Opsware® SAS Administration Guide*.

# Chapter 9: Opsware SAS Uninstallation

**IN THIS CHAPTER**

This section discusses the following topics:

- Overview of Uninstalling Opsware SAS
- Procedures for Uninstalling Cores

## Overview of Uninstalling Opsware SAS

You might need to uninstall an Opsware core in the following scenarios:

- You have an Opsware core in a lab setting before installing Opsware SAS in a production environment. You might want to uninstall the Opsware core after you finish testing it.
- You are consolidating facilities and want to uninstall an Opsware core in one facility in preparation to moving it to another facility.

Uninstalling the Model Repository permanently deletes all data in the database. But when you uninstall an Opsware core, you can choose to preserve the Opsware SAS data in the Model Repository database. If you choose to preserve this data, the Opsware Installer stops the uninstallation.

Stopping the uninstallation gives you the opportunity to back up the data in the Model Repository. After you begin the Model Repository uninstallation, the Opsware Installer will not preserve any data in the Model Repository.

You can also choose to preserve or remove all the packages stored on the Software Repository.

You can also choose to preserve the database of cryptographic material for the Opsware core. If you choose to preserve crypto, the database of cryptographic material will be saved; otherwise it will be deleted when the uninstallation finishes.

Before you uninstall an Opsware core, Opsware Inc. recommends that you back up the Oracle database running on the server where the Model Repository is installed. See your Oracle documentation for the steps required to back up an Oracle database.

## Procedures for Uninstalling Cores

This section discusses the following topics:

- Uninstalling a Standalone Core
- Uninstalling One Core in a Multimaster Mesh
- Uninstalling an Entire Multimaster Mesh of Opsware Cores
- Decommissioning a Facility in the Opsware Command Center

### Uninstalling a Standalone Core

Perform the following steps to uninstall a standalone core:

1. Before you uninstall the Opsware core components from the servers running them, you should deactivate the servers in the Opsware Command Center. Otherwise, if you try to re-install an Opsware core component on one of the servers later, the installation will fail. (For more information, see "Deactivating a Server" in the *Opsware® SAS User's Guide*.)
2. Log in as root.
3. Change to the root directory:
   ```
   cd /
   ```
4. Run the `uninstall_opsware.sh` script:
   ```
   /opsware_system/opsware_installer/uninstall_ opsware.sh -r
   <response-file>
   ```
5. At the components prompt, select one or more components to uninstall:
   ```
   Welcome to the Opsware Installer.
   Please select the components to uninstall.
   1 ( ) Opsware Gateway
   2 ( ) OS Provisioning Build Manager
   3 ( ) OS Provisioning Media Server
   4 ( ) OS Provisioning Boot Server
   5 ( ) Opsware Command Center (OCC)
   6 ( ) Opsware Documentation
   ```

```
7 ( ) Opsware Global Filesystem Server (OGFS)
8 ( ) Software Repository (word)
9 ( ) Command Engine (way)
10 ( ) Data Access Engine (spin)
11 ( ) Model Repository (truth)
```

If the Opsware Gateway does not run on a separate server, uninstall it last.

6 Remove the `/var/lc/install_opsware` directory.

If you indicated at the prompt that you want to preserve crypto (the database of cryptographic material), you should not delete the `/var/lc` directory. Deleting the `/var/lc` directory deletes the database of cryptographic material.

### Uninstalling One Core in a Multimaster Mesh

When uninstalling a core from a multimaster mesh, you should not uninstall the source core unless you are planning to uninstall the entire mesh.

See "Uninstalling an Entire Multimaster Mesh of Opsware Cores" on page 139 in this chapter for more information.

Perform the following steps to uninstall one core in a multimaster mesh:

1 Log into any Opsware Command Center that is still online to perform the following tasks:

1. Using the System Configuration feature, update the `listeners` configuration parameter by removing the entry for the core that is being uninstalled. Update the listeners parameter by selecting "Model Repository, Multimaster Component" in the System Configuration page.
2. If a Data Access Engine that is being uninstalled is currently serving as the multimaster central role, a Data Access Engine in another core must be selected to serve as Multimaster Central.

   See "Reassigning the Data Access Engine to a Secondary Role" in the *Opsware® SAS Administration Guide*.
3. Verify that all transactions have propagated to the other facilities, except for the facility that is being uninstalled.

   See "Verifying Multimaster Transaction Traffic" on page 106.

2. Decommission the facility for the core you are uninstalling. See “Decommissioning a Facility in the Opsware Command Center” on page 139.

3. Restart the Model Repository Multimaster Component in all cores except the core that is being uninstalled by entering the following command as root on the server running the engine:

   ```
   /etc/init.d/vaultdaemon stop
   ```

   ```
   /etc/init.d/vaultdaemon start
   ```

4. Stop the Opsware Command Center in the core that is being uninstalled by entering the following command as root:

   ```
   /etc/init.d/occ.server stop
   ```

5. In the core that is being uninstalled, stop all Data Access Engines.

   Log in as root to the server where the Data Access Engine is running and enter the following command:

   ```
   /etc/init.d/spin stop
   ```

6. If the Opsware Command Center and the Data Access Engine are installed on different servers, you must also run the `spin stop` command on the Opsware Command Center server.

7. Stop the Model Repository Multimaster Component in the core that is being uninstalled by entering the following command as root on the server running the engine:

   ```
   /etc/init.d/vaultdaemon stop
   ```

8. Restart the Data Access Engine that is serving as Multimaster Central by entering the following command as root:

   ```
   /etc/init.d/spin stop
   ```

   ```
   /etc/init.d/spin start
   ```

9. For the core that you are uninstalling, on each server running an Opsware component, run the following script.

   ```
   /opsware_system/opsware_installer/uninstall_opsware.sh
   ```

   Uninstall the components by following the instructions in step 4 through step 6 in the section “Uninstalling a Standalone Core.”

### Uninstalling an Entire Multimaster Mesh of Opsware Cores

Perform the steps in this procedure only when you want to uninstall all cores in a multimaster mesh:

1. Stop the Opsware Command Center by logging on as root to the server where the Opsware Command Center is running and enter the following command:

   ```
   /etc/init.d/occ.server stop
   ```

2. Stop the Data Access Engine.

   Log in as root to the server where the Data Access Engine is running and enter the following command:

   ```
   /etc/init.d/spin stop
   ```

   If the Opsware Command Center and the Data Access Engine are installed on different servers, you must also run the `spin stop` command on the Opsware Command Center server.

3. Stop the Model Repository Multimaster Component in all cores by logging in to the servers running the engines and entering the following command as root:

   ```
   /etc/init.d/vaultdaemon stop
   ```

4. In each core, uninstall the Opsware components on the servers where they are installed.

   ```
   /opsware_system/opsware_installer/uninstall_opsware.sh
   ```

   Follow the instructions in step 4 through step 6 in the section “Uninstalling a Standalone Core.”

### Decommissioning a Facility in the Opsware Command Center

Performing this procedure does not shut down or uninstall Opsware SAS in a facility. Decommission facilities with care, because this task cannot be undone.

When you decommission a facility, the facility is still listed in the Opsware Command Center, however, it is grayed out. After a short name is used, even if it is decommissioned, that name cannot be reused.

Perform the following steps to decommission a facility:

1. In the Opsware Command Center, deactivate the server running the core of the facility that you wish to decommission. (For instructions, see "Deactivating a Server" in the *Opsware® SAS User's Guide*.)
2. From the navigation panel, click Environment ➤ Facilities. The Facilities page appears.
3. Select the facility that you want to decommission.
4. On the Properties tab, note the answer to the following question:

   ```
   Is this facility in use?
   ```

   If the answer is No, the **Decommission** button is displayed.
5. Click **Decommission**.

# Appendix A: Oracle Setup for Model Repository

**IN THIS APPENDIX**

This section discusses the following topics:

- Supported Oracle Versions
- Setting Up the Database
- Database Monitoring for the Model Repository

The Opsware Model Repository stores information in an Oracle database. Before running the Opsware Installer, you must create an Oracle database on the server where you will install the Model Repository.

## Supported Oracle Versions

Support for the Model Repository is limited to specific versions of Oracle running on specific versions of operating systems. Table A-1 lists the supported Oracle versions.

*Table A-1: Supported Oracle Versions for Model Repository*

| ORACLE EDITION | VERSIONS |
|---|---|
| Oracle Standard Edition | 9.2.0.4.0<br>9.2.0.6.0 |
| Oracle Enterprise Edition | 9.2.0.4.0<br>9.2.0.6.0 |

Oracle version 9.2.0.5.0 is not supported with this release of Opsware SAS.

To be supported on the Model Repository, the Oracle versions listed in Table A-1 are limited to the operating systems listed in Table A-2.

*Table A-2: Supported Operating Systems for Model Repository*

| SUPPORTED OPERATING SYSTEMS FOR MODEL REPOSITORY | VERSIONS | ARCHITECTURE |
|---|---|---|
| Sun Solaris | Solaris 8<br>Solaris 9 | Sun SPARC<br>Sun SPARC |
| Red Hat Linux | Red Hat Enterprise Linux 3 AS | 32 bit x86 |

## Setting Up the Database

To set up the database, perform the following steps:

1. On the server where you will also install the Model Repository, install one of the supported versions of Oracle. Before installing Oracle, make sure that the operating system is supported by Opsware SAS.

   If you want the Model Repository to access a remote Oracle database, install Oracle Client on the server where you will install the Model Repository. The `truth.orahome` parameter (ORACLE_HOME) you specify during the installation of the Model Repository must point to the location of the Oracle Client.

2. Create an Oracle database with the UTF8 character set.

   The following clauses in the `CREATE DATABASE` statement include recommended sizes. (Your organization's guidelines might specify different sizes.)

   ```
   DEFAULT TEMPORARY TABLESPACE TEMP TEMPFILE SIZE 1000M
   AUTOEXTEND ON NEXT 64M MAXSIZE UNLIMITED

   UNDO TABLESPACE "UNDO" DATAFILE SIZE 1000M AUTOEXTEND ON NEXT
   64M MAXSIZE UNLIMITED
   ```

   Storage requirements for the database grow as the number of managed servers grows. As a benchmark figure, you should allow an additional 3.1 GB database storage for every 1000 servers in the facility that Opsware SAS manages.

3 Specify initialization parameters in the `init.ora` file. Opsware SAS requires the following parameter settings. (All other settings can follow your organization's guidelines, or you can use the default settings.)

```
optimizer_mode = choose
query_rewrite_enabled = true
query_rewrite_integrity = trusted
open_cursors >= 2000
shared_pool >= 200000000
sort_area_size >= 1048576
nls_sort = generic_m
job_queue_processes >= 2
processes > 1000
db_block_size >= 8192
java_pool_size >= 50000000
workarea_size_policy = auto
sessions > 1000
cursor_sharing = similar
pga_aggregate_target >= 268435456
nls_length_semantics = char
```

Set the `nls_length_semantics` parameter to `char` for a standalone core installation. When you are adding a core to a multimaster mesh, set `nls_length_semantics` to the same value that the other Opsware cores are using. If you use different settings in the cores, Opsware SAS will not function correctly. Contact Opsware Professional Services for assistance upgrading the setting for an `nls_length_semantics` parameter in a core.

4 Set up the `tnsnames.ora` file.

The `tnsnames.ora` file enables resolution of database names used internally by Opsware SAS. A `tnsnames.ora` file is required on the core servers running the following components: Model Repository, Data Access Engine, Web Services Data Access Engine, and Model Repository Multimaster Component.

In a standalone core, the `tnsnames.ora` file must contain an entry for the Model Repository. For example:

```
truth =
(DESCRIPTION=
(ADDRESS=(HOST=magenta.opsware.com)(PORT=1521)
(PROTOCOL=tcp))
(CONNECT_DATA=(SERVICE_NAME=truth)))
```

In a multimaster mesh, the `tnsnames.ora` file of the central (master) core must contain an entry for its own Model Repository. The file must also have entries for the Model Repositories of the other cores in the mesh. For the entries of the other (non-central) cores, the host specifies the central core Gateway, and the port number is derived from this formula: (20000) + (facility ID of the non-central core).

The following `tnsnames.ora` example is for the central core of a multimaster mesh. In this example, the TNS service name of the central core is `orange_truth`, which runs on the host `orange.opsware.com`. The TNS name of the non-central core is `cyan_truth`, which has a facility ID of 556. Note that the entry for `cyan_truth` specifies `orange.opsware.com`, the host running the central core's Gateway.

```
orange_truth =
(DESCRIPTION=
(ADDRESS=(HOST=orange.opsware.com)(PORT=1521)(
PROTOCOL=tcp))
(CONNECT_DATA=(SERVICE_NAME=truth)))
cyan_truth =
(DESCRIPTION=(ADDRESS=(HOST=orange.opsware.com)(PORT=20556)
(PROTOCOL=tcp))
(CONNECT_DATA=(SERVICE_NAME=truth)))
```

In a multimaster mesh, the `tnsnames.ora` file of a non-central (non-master) core must contain an entry for its own Model Repository, but does not require entries for other cores in the mesh. In the following `tnsnames.ora` example, the TNS service name of the non-central core is cyan_truth, and the core runs on the host, cyan.opsware.com.

```
cyan_truth =
(DESCRIPTION=
(ADDRESS=(HOST=cyan.opsware.com)(PORT=1521)
(PROTOCOL=tcp))
(CONNECT_DATA=(SERVICE_NAME=truth)))
```

If you install the Opsware core on multiple servers, the `tnsnames.ora` file with the same directory path must exist on the servers where the following Opsware components are installed: Model Repository, Data Access Engine, Opsware Command Center, Opsware Global File System, Model Repository Multimaster Component.

5. Start the Oracle listener.

6. Initialize the Oracle JVM. The Oracle Installer provides an option for this, but you can also use the following script in the Oracle product directory:
   `$ORACLE_HOME/javavm/install/initjvm.sql`

7 Create the following tablespaces:

```
AAA_DATA
AAA_INDX
LCREP_DATA
LCREP_INDX
TRUTH_DATA
TRUTH_INDX
```

When you create the DATA tablespaces, you should use the sizes shown in the following example:

```
CREATE TABLESPACE "LCREP_DATA" LOGGING DATAFILE SIZE 1000M
AUTOEXTEND ON NEXT 64M MAXSIZE UNLIMITED EXTENT MANAGEMENT
LOCAL SEGMENT SPACE MANAGEMENT AUTO;
```

When sizing the tablespaces, follow the general guidelines shown in the following table. If you need to determine a more precise tablespace sizing, contact your Opsware, Inc. Support Representative.

*Table A-3: Tablespace Sizes*

| TABLESPACE | MB/1000 SERVERS | MINIMUM SIZE |
|---|---|---|
| AAA_DATA | 256 MB | 256 MB |
| AAA_INDX | 256 MB | 256 MB |
| LCREP_DATA | 3,000 MB | 1,500 MB |
| LCREP_INDX | 1,600 MB | 800 MB |
| TRUTH_DATA | 1,300 MB | 700 MB |
| TRUTH_INDX | 300 MB | 400 MB |

8 Create the opsware_admin database user.

Opsware SAS uses the opsware_admin user to install and manage the Model Repository. Use the TRUTH_DATA tablespace with unlimited quota as the default tablespace for the opsware_admin user. Set the temporary tablespace according to your organization's guidelines.

9 Grant privileges to the opsware_admin user as shown in the following SQL statements:

```
grant alter session to opsware_admin with admin option;
grant create procedure to opsware_admin with admin option;
grant create public synonym to opsware_admin with admin
option;
```

```
grant create sequence to opsware_admin with admin option;
grant create session to opsware_admin with admin option;
grant create table to opsware_admin with admin option;
grant create trigger to opsware_admin with admin option;
grant create type to opsware_admin with admin option;
grant create view to opsware_admin with admin option;
grant delete any table to opsware_admin with admin option;
grant drop public synonym to opsware_admin with admin option;
grant select any table to opsware_admin with admin option;
grant select_catalog_role to opsware_admin with admin
option;
grant query rewrite to opsware_admin with admin option;
grant restricted session to opsware_admin with admin option;
grant execute on dbms_utility to opsware_admin with grant
option;
grant analyze any to opsware_admin;
grant select, insert, update, delete on sys.aux_stats$ to
opsware_admin;
grant alter system to opsware_admin;
grant create role to opsware_admin;
grant create user to opsware_admin;
grant alter user to opsware_admin;
grant drop user to opsware_admin;
grant create profile to opsware_admin;
grant alter profile to opsware_admin;
grant drop profile to opsware_admin;
```

**10** Set the `NLS_LANG` environment variable for the `oracle` Unix user.

This environment variable is required for the export and import operations when installing a multimaster core. The syntax of `NLS_LANG` follows:

```
NLS_LANG=<languague>_<territory>.<client_characterset>
```

For example, in the United States you might set `NLS_LANG` to the following value:

```
NLS_LANG=AMERICAN_AMERICA.UTF8
```

The value of `NLS_LANG` must match the character set used by the database, which can be determined by the following query:

```
sql> select value from nls_database_parameters where
parameter='NLS_CHARACTERSET';
```

If the export (source) and import (target) databases have different character sets, then for both set `NLS_LANG` to the character set of the export database.

**11** Set up database monitoring. (See the following section.)

## Database Monitoring for the Model Repository

For the Oracle instance that the Opsware Model Repository uses, you should set up monitoring for the following key diagnostics:

- The availability of the Oracle instance, database, and listener process.
- The availability of space for the Model Repository (truth) schema growth

Additionally, Opsware Inc. recommends that you monitor key Oracle log files, including the `alert.log` and background and user trace files.

### Instance and Database Availability

In this topic, the examples for basic monitoring assume that the Oracle instance name is `truth`.

Opsware SAS becomes unavailable when Oracle becomes unavailable. Therefore, to ensure that Opsware SAS has access to the Oracle database, you must ensure that the Oracle instance is running, the Oracle database is open, and the listener is monitoring for connections.

#### *Checking the Instance*

To check the Oracle instance, perform the following steps:

1. To check for the status, login as the `oracle` Unix user and use the `ps` command to look for the processes with names starting with `ora_`. For example:

```
oracle$ ps -ef | grep ora_
oracle 14239     1  0   Mar 19 ?        0:08 ora_lgwr_truth
oracle 14245     1  0   Mar 19 ?        0:00 ora_reco_truth
oracle 14241     1  0   Mar 19 ?        0:16 ora_ckpt_truth
oracle 14237     1  0   Mar 19 ?        0:04 ora_dbw0_truth
oracle 14243     1  0   Mar 19 ?        0:16 ora_smon_truth
oracle 14235     1  0   Mar 19 ?        0:00 ora_pmon_truth
oracle 14247     1  0   Mar 19 ?        0:00 ora_cjq0_truth
```

2. Confirm that the instance is running by connecting to the database as `sysdba`. (Be sure to set your `ORACLE_HOME` and `ORACLE_SID` environment variables appropriately.)

```
oracle$ sqlplus "/ as sysdba"
. . .
Connected to:
Oracle9i Enterprise Edition Release 9.2.0.4.0 - Production
JServer Release 9.2.0.4.0 - Production
```

The "`Connected to:`" message confirms that the instance is available.

### Checking the Database

Opsware SAS needs the database to be mounted and open for general use in order to function. To check the database, perform the following steps:

1. To check the status of the database, connect to the instance as `sysdba` and issue the following query:

   ```
   sql> select database_status from v$instance;
   ```

   The result should be `ACTIVE`.

2. To check the mode in which the database was opened, issue the following query:

   ```
   sql> select open_mode from v$database;
   ```

   The result should be `READ WRITE`.

### Checking the Listener

To check the Oracle listener (`tnslsnr`), perform the following steps:

1. Check the status of the listener with the `lsnrctl` command:

   ```
   oracle$ lsnrctl status
   . . .
   Service "truth" has 1 instance(s).
     Instance "truth", status READY, has 1 handler(s) for this
   service...
   ```

   The status should be `READY`.

2. Test connectivity to the instance from the Data Access Engine (spin) and Web Services Data Access Engine (twist) hosts by running the `tnsping` utility (or by connecting with SQL*Plus with a net-service name identifier):

   ```
   oracle$ tnsping truth
   . . .
   Attempting to contact
   (DESCRIPTION=(ADDRESS=(HOST=localhost)(PORT=1521)(PROTOCOL=t
   cp))(CONNECT_DATA=(SERVICE_NAME=truth)))
   OK (0 msec)
   ```

   The `OK` statement confirms that the listener is up and can connect to the instance.

### Checking for Datafile Space Availability

Opsware SAS stores its data in a series of size tablespaces, each consisting of one or more datafiles. For the size of the data set to grow, you must ensure that each tablespace has enough space for the allocation of new rows.

You can verify the auto-extensibility of tablespaces with the following query:

```
sql> select d.file_name, d.tablespace_name, d.status,
```

```
d.autoextensible,
d.bytes / 1024 / 1024,
nvl(d.bytes - sum(s.bytes), d.bytes) / 1024 / 1024,
nvl(d.bytes - sum(s.bytes), d.bytes) / d.bytes * 100
from sys.dba_data_files d, sys.dba_free_space s
where (s.file_id (+)= d.file_id)
and d.bytes is not null
group by d.tablespace_name, d.file_name, d.status,
d.autoextensible, d.bytes;
```

You can also monitor tablespace usage by running a test for the System Diagnosis feature of the Opsware Command Center. The test is named Oracle Tablespaces and is listed under the Data Access Engine component. This test checks to see if manually-extended tablespaces are less than 85% full.

### Monitoring Oracle Log Files

Monitor the following Oracle log files:

- The Oracle `alert.log` file. (Check this file for `ORA-` errors because some of the errors will not be displayed.)

  `$ORACLE_BASE/admin/truth/bdump/alert_truth.log`

  `$ORACLE_BASE/admin/truth/[bcu]dump/*.trc`

Configure a `cron` job to perform the following actions:

- Periodically poll for changes to files, for the creation of files, or for the presence of `ORA-` errors.
- Report these errors by e-mail or another way to a DBA.

# Appendix B: TIBCO Rendezvous Configuration for Multimaster

**IN THIS APPENDIX**

This section discusses the following topics:

- TIBCO Rendezvous and Opsware SAS
- TIBCO Rendezvous Configuration

## TIBCO Rendezvous and Opsware SAS

In a multimaster mesh, Opsware SAS uses the TIBCO Certified Messaging system to synchronize the Model Repositories in different facilities.

The Opsware Installer automatically installs and configures TIBCO Rendezvous. By default, the installer configures the Rendezvous neighbors in a star topology, in which the source core is at the center. Unless you want another configuration, no further action is required by you.

## TIBCO Rendezvous Configuration

This section explains how to add TIBCO routers and neighbors. For more information, see the following TIBCO Rendezvous documentation:

- *TIBCO Rendezvous Installation Guide*
- *TIBCO Rendezvous Concepts*

### Running the TIBCO Rendezvous Web Client

To run the TIBCO Rendezvous web client, enter the following URL in a web browser:

```
http://<hostname>:7580
```

The <hostname> is the IP address or fully-qualified host name of the server running the Model Repository Multimaster Component (vault). The TIBCO Rendezvous General Information page appears.

### Adding a TIBCO Router

To add a TIBCO router, perform the following steps:

1. Run the TIBCO Rendezvous web client.
2. From the left navigation panel, click Routers under Configuration. The Routers Configuration page appears.
3. Make sure that your browser can resolve the host name so that the link in the Router Name field functions correctly.
4. In the Router Name field, enter a value. Usually, you enter the facility name for the router name.
5. Click **Add Router**. The new router appears in the table on the page.
6. In the Local Network column under Interfaces, click the number link for the router you just added. The Local Network Interfaces Configuration page appears.
7. Define a new network by entering the following data:
   1. In the Local Network Name field, enter the network name. In most cases, the network is given the same name as the facility name.
   2. In the Service field, set the service to 7500.
   3. Click **Add Local Network Interface**. The new local network appears in the table in the page.
8. Click the link for the new local network name. The Subject Configuration page appears.
9. In the Subject field, enter a greater-than symbol (>) and click **Import** and **Export**. (The greater-than symbol means “any.”) The greater-than symbol appears in the Import Subjects and Export Subjects tables in the page.
10. Repeat the previous steps for the other facilities in the multimaster mesh.

### Adding a TIBCO Rendezvous Neighbor

To add a TIBCO Rendezvous neighbor, perform the following steps:

1. In the core Gateway properties file, add the following line:
   `opswgw.ForwardTCP=<port>:<remote_realm>:<remote_host>:7501`
   The <port> is derived from this formula: 10000 + remote_facility_ID. The <remote_realm> is the realm name of the core Gateway in the remote facility. The <remote_host> is the IP address of the server running the Model Repository Multimaster Component (vault) in the remote facility. In the following example, the remote facility ID, is 667, the realm name is LIME, and the IP address of the Model Repository Multimaster Component is 192.168.165.98:
   `opswgw.ForwardTCP=10667:LIME:192.168.165.98:7501`
2. Run the TIBCO Rendezvous web client.
3. From the left navigation panel, click Routers under Configuration. The Routers Configuration page appears.
4. In the Neighbor column of the table, click the number link for the router you added in the previous procedure. The Neighbor Interfaces Configuration page appears. You must define a neighbor for each facility in the multimaster mesh, except for the local facility.
5. In the Host field under the Remote Endpoint section, enter the host name of the server running the local core Gateway.
6. In the Port field under the Local Endpoint section, enter 7501.
7. In the Port field under the Remote Endpoint sections, set the port to the value derived from the following formula: 10000 + remote_facility_ID.
8. In the Router Name field under the Remote Endpoint section, enter the router name for the other facility.
9. For the Connection Type, select Normal Connection.
10. Click **Add Neighbor Interface**. The Local and Remote endpoints are added to the table in the page.

### Verifying TIBCO Rendezvous Configuration

To see if the neighbor has connections to a facility, perform the following steps:

1. Run the TIBCO Rendezvous web client.

2 Click Connected Neighbors in the left navigation panel. For each neighbor you defined for this facility, you should see links for the rvrd interface.

# Appendix C: Opsware Gateway Properties File

**IN THIS APPENDIX**

This section discusses the following topics:

- Syntax of the Opsware Gateway Properties File
- Options for the opswgw Command

## Syntax of the Opsware Gateway Properties File

An Opsware Gateway properties file can have the following entries:

`opswgw.Gateway=name`

(Required) Set the name of the Opsware Gateway. This name must be unique in a Gateway network.

`opswgw.Realm=realm`

(Required) All Opsware Gateways operate in a named realm. A realm is an abstract name given to the collection of servers which are serviced by the Gateways in the realm. Realms can support an IP address space which may overlap with another realm. Realms are also used to define bandwidth utilization constraints on Opsware SAS functions in that realm.

`opswgw.Root=true | false`

Indicates that this Gateway should act as a root of the Gateway network. All Gateways in a root realm must be root Gateways. The default is false.

`opswgw.Daemon=true | false`

Daemonize the process. The default is false.

`opswgw.Watchdog=true | false`

Start an internal watchdog process to restart the Gateway in case a failure or a signal. A `SIGTERM` sent to the watchdog will stop the watchdog and Gateway processes. The default is false.

`opswgw.HardExitTimeout=seconds`

The number of seconds the main thread will wait (after a restart or exit request) for internal threads and queues to quiesce before a hard exit is performed.

`opswgw.LogLevel=INFO | DEBUG | TRACE`

Set the logging level. The `DEBUG` and `TRACE` produce a lot of output which will only be relevant to developers. The default is `INFO`.

`opswgw.LogFile=file`

The basename of the log file.

`opswgw.LogNum=num`

The number of rolling log files to keep.

`opswgw.LogSize=size`

The size in bytes of each log file.

`opswgw.TunnelDst=[lip1:]lport1[:crypto1],...`

Start up a tunnel destination listener. The tunnel listener can listen on a list of ports (a comma-separated list with no spaces.) If the port is prefixed with an IP, then the listener will only bind to that IP address. Examples: `2001, 10.0.0.2:2001, 2001:/var/foo.pem, 10.0.0.2:2001:/var/foo.pem`

`opswgw.TunnelSrc=rhost1:rport1:cost1:bw1[:crypto1],...`

Create a tunnel between this Gateway and the Gateway listening at `rhost1:rport1`. The link `cost1` and link bandwidth `bw1` must be set. The cost is a 32bit unsigned int, and bandwidth is in Kbits/sec (K=1024bits). (Additional tunnels are separated by commas.) Examples: `gw.foo.com:2001:1:0, gw.bar.com:2001:10:256:/var/foo.pem`

`opswgw.TunnelTCPBuffer=bytes`

Set the size TCP send and recv buffer to `bytes`. The system's OS must be configured to handle this value. View the Gateway's log file to see if the value given here will work on the current system.

`opswgw.ValidatePeerCN=true | false`

Indicates whether the peer CN be validated. The peer needs to be turned off during the installation of an untrusted Gateway. The default is true.

`opswgw.ProxyPort=[lip1:]lport1,[lip2:]lport2,...`

The SSL proxy listen port. If more than one proxy listen port is needed, add more using a comma separated list.

`opswgw.ForwardTCP=[lip1:]lport1:realm1:rhost1:rport1,...`

Create a static TCP port forward. Forward the local port `lport` to the remote service `rhost:rport`, which is in `realm`. A blank `realm` (e.g., `lport::rhost:rport`) means route to the root realm.

`opswgw.ForwardUDP=[lip1:]lport1:realm1:rhost1:rport1,...`

Create a static UDP port forward. Forward local port `lport` to remote service `rhost:rport`, which is in `realm`. If `realm` is blank (e.g., `lport::rhost:rport`) it means route to the root realm. (Warning: Some UDP services, such as DHCP, cannot be proxied in this manner.)

`opswgw.GWAddress=lhost`

Set the local host address (IP or name) that this Gateway uses to tell other components how to contact it. This value is used by the core to discover new core-side Gateways. It is also used to communicate the active list of Gateways that are servicing a realm to proxy clients (such as Agents) via the `X-OPSW-GWLIST` mime header.

`opswgw.IdentPort=[lip:]lport`

Start up an ident service listening on local port `lport`.

`opswgw.FinalizeTCPPortMap=true|false`

If true, remove the TCP source port from the ident port map immediately before the socket is closed. If false, the mapping persists until the port is reused. Warning: Only use false if you know what you are doing. The default is true.

`opswgw.FinalizeUDPPortMap=true|false`

If true, remove the UDP source port from the ident port map immediately before the socket is closed. If false, the mapping persists until the port is reused. Warning: Only use false if you know what you are doing. The default is true.

`opswgw.AdminPort=[lip:]lport[:crypto1]`

Start up an administration interface listening on local port `lport`, which is optionally bound to the local IP `lip`. If crypto is desired, then include a crypto specification file name.

`opswgw.ConnectionLimit=int`

The soft memory tuning limit of maximum number of connections.

`opswgw.OpenTimeout=seconds`

Only wait this many seconds for a remote `CONNECT` call to establish a remote connection.

`opswgw.ConnectTimeout=seconds`

Only wait this many seconds for the `connect()` to complete. If a timeout occurs, then an HTTP 503 message is returned to the client (via the ingress Gateway). The client will get this message if the `ConnectTimeout` plus the Gateway mesh transit delay is less than the `OpenTimeout`.

`opswgw.ReorderTimeout=seconds`

In the event of out-of-order messages (for a TCP flow), limit the amount of time to wait for messages (needed for reassembly) to arrive.

`opswgw.QueueWaitTimeout=seconds`

Maximum time that a tunnel message can wait at the head of an internal routing queue (while waiting for a tunnel to be restored).

`opswgw.LsaPublishRate=seconds`

Send the Link State Advertisements (LSAs) every X seconds.

`opswgw.LsaExtendRate=count`

Send an extended LSA for every count number of normal LSAs. Example: If `LsaPublishRate` is 10.0 seconds and `LsaExtendRate` is 30, then every 30 LSAs (about every 300 seconds) an extended LSA is published.

`opswgw.LsaTTLMultiple=float`

Set the TTL for LSAs to this number multiplied by the `LsaPublishRate`. Example: If `LsaPublishRate` is 10 seconds and `LsaTTLMultiple` is 3 then, the TTL for LSAs published by this Gateway is set to 30 seconds.

`opswgw.LsaExtendTTLMultiple=float`

Set the TTL for extended LSAs to this number multiplied by the `LsaPublishRate` and the `LsaExtendRate`. Example: If the `LsaPublishRate` is 15 seconds and the `LsaExtendRate` is 30 and the `LsaExtendTTLMultiple` is 8, then the TTL for extended LSA information is 3600 seconds (because 15 * 30 * 8 = 3600). One function of the in-memory database of the extended LSA information is to form the `X-OPSW-GWLIST` MIME header.

`opswgw.MaxRouteAge=seconds`

Discard the routes from the routing table that have not been refreshed within this number of seconds.

`opswgw.TunnelTimeoutMultiple=float`

This number, multiplied by the `LsaPublishRate`, gives the maximum time that a tunnel can be idle before it is garbage collected.

`opswgw.DoNotRouteService=host1:port1,host2:port2,...`

If a local client creates a proxy connection to `host:port`, then do not route the message; service it locally. This is used to handle certain services locally in the Gateway's current realm.

`opswgw.ForceRouteService=`
`host1:port1:realm1,host2:port2:realm2,...`

If local client creates a proxy connection to `host:port`, then force the message to route to realm.

`opswgw.HijackService=host1:port1,host2:port2,...`

If the local Gateway sees a connection to host:port via a tunnel, and the source realm is different than the local realm, then service the connection. Otherwise, let the message continue to its destination. This feature is useful for implementing transparent caches.

`opswgw.EgressFilter=tcp:dsthost1:dstport1:srchost1:srcrealm1,...`

If the local Gateway sees a `tcp` connection attempt to `dsthost:dstport` from `srchost1:srcrealm1`, then allow the connection. The implied default is to deny all connections. If you want to allow all traffic, then specify `*:*:*:*:*`. Watch out for shell quoting. It is common for an egress filter to only allow connections from the root realm. This can be expressed by leaving the `srcrealm` blank. Example: `tcp:10.0.0.5:22:172.16.0.5:` would allow tcp connections to 10.0.0.5, port 22, from 172.16.0.5 in a root realm.

`opswgw.IngressMap=ip1:name,ip2:name,...`

When sending an open message (and the `srcip` is in the ingress map), append (as metadata) the `ip:name` mapping to the open message. This allows a remote egress filter to use the `name` as the `srchost` instead of the `ip`. This feature supports the addition of a server to a farm without the need to add the server to many `EgressFilter` entries.

`opswgw.LoadBalanceRule=`
`tcp:thost:tport:mode:rhost1:rport1:rhost2:rport2, ...`

When receiving an open connection message for `thost:tport`, load balance the connection over real hosts `rhost1:rport1, rhost2:rport2` etc. The load balance strategy is defined by `mode`. There is currently only one mode: `STICKY`. This mode does sticky load balancing based on a hash of the source realm and `ip`. Remember to add an egress filter for `thost:tport`. You do not need to add egress filters for the targets. Load balancing is only for tcp connections.

`opswgw.LoadBalanceRetryWindow=seconds`

If an error occurs when using a load balanced target (e.g., `rhost1:rport1` above) then the target is marked `in-error`. This parameter controls how many seconds a Gateway will wait until it re-tries the target. If the target is missing (i.e., an `RST` is received upon the connection request) the load balancer will silently try to find a good target.

`opswgw.MinIdleTime=seconds`

The minimum number of seconds a connection can be idle, during an overload condition, before it will be considered for reaping.

`opswgw.GCOverloadTrigger=float`

The fraction of `SoftConnectionLimit` at which to start overload protection measures. When the number of open connections hits this overload trigger point, the overload protection kicks in, reaping the most idle connections over `MinIdleTime`. Overload protection quits when the connection count falls below the overload trigger point.

`opswgw.GCCloseOverload=true | false`

When a client tries to open a connection after the `ConnectionLimit` has been reached, this property tells the Gateway what to do with the new connection. A value of true causes the Gateway to close the new connection. A value of false causes the Gateway to park the new connection in the kernel's backlog and to service it once the overload condition subsides. The proper setting is application dependent. The default is false.

`opswgw.VerifyRate=seconds`

When a connection stops moving data for this number of seconds, a connection verify message is sent to the remote Gateway to check that the connection is still open on its end. This check is repeated periodically and indefinitely when the timeout has expired.

`opswgw.OutputQueueSize=slots`

The size of the tunnel output queues. These queues store messages destined for remote Gateways. Each remote Gateway has an output queue.

`opswgw.DefaultChunkSize=bytes`

The default (maximum) IO chunk size when encapsulating a TCP stream. This default is only used on links with no bandwidth constraint.

`opswgw.LinkSaturationTime=seconds`

On links with a bandwidth constraint, the chunk size (see `DefaultChunkSize`) is computed based on two parameters. The first is the link's bandwidth constraint. The second is the amount of time that the bandwidth shaper should utilize the full, real, bandwidth on the link. This parameter controls the duty cycle of the bandwidth shaper. Smaller values give a smoother bandwidth control at the cost of more overhead, because each smaller IO chunk has a header.

`opswgw.MaxQueueIdleTime=seconds`

The maximum time to keep an idle output queue before garbage collection removes it.

`opswgw.TunnelPreLoad=slots`

The maximum number of output queue slots to use before waiting for the first `Ack` message. This allows for pipelining in Long Fat Pipes. This value is reduced geometrically to one as the number of queue slots diminish.

`opswgw.BandwidthAveWindow=samples`

The maximum number of IO rate samples for the bandwidth estimation moving window. The samples in this window are averaged to provide a low pass estimate of the bandwidth in use by a tunnel. This estimate has high frequency components due to the sharp edge of the filter window.

`opswgw.BandwidthFilterPole=float`

The pole of a discrete-time first-order smoothing filter used to remove the high frequency components of the moving window estimator. Set the value to 0.0 to turn off this filter.

`opswgw.StyleSheet=URL`

Add a stylesheet link to URL when rendering the admin UI. This is useful for embedding the admin UI in another web-based UI. In addition to using this property to control the default stylesheet, a dynamic stylesheet override is supported by adding the variable `StyleSheet=¡url¿/style.css` to the admin UI URL.

`opswgw.PropertiesCache=file`

Link cost and bandwidth can be controlled via parameter-modify messages over the tunnel connections. These real-time adjustments are made to the running process and written to a parameter cache which will override the properties file or command line arguments.

## Options for the opswgw Command

All of the properties in the preceding section can be specified as options for the `opswgw` command. For example, the `opswgw.Gateway=foo` entry in the properties file is equivalent to the following command-line option:

```
/opt/OPSWgw/bin/opswgw --Gateway foo
```

Command-line arguments override corresponding entries in the properties file. In addition to the entries listed in the preceding section, the `opswgw` command can specify a properties file as follows:

```
/opt/OPSWgw/bin/opswgw --PropertiesFile file
```

# Index

## P



## R



## S

## T



## U



## V

