Oracle® Secure Backup

Installation and Configuration Guide

Release 10.4

E21477-05

April 2013

How to install, uninstall, and manage hardware and network configuration in Oracle Secure Backup

ORACLE®

Oracle Secure Backup Installation and Configuration Guide, Release 10.4

E21477-05

Primary Author: Padmaja Potineni

Contributing Authors: Lance Ashdown, Craig B. Foch

Contributors: Anand Agrawal, Tammy Bednar, George Claborn, Michael Chamberlain, Sumit Chougule, Donna Cooksey, Rhonda Day, Senad Dizdar, Tony Dziedzic, Judy Ferstenberg, Steven Fried, Geoff Hickey, Ashok Joshi, Cris Pedregal-Martin, Chris Plakyda, George Stabler, Radhika Vullikanti, Joe Wadleigh, Steve Wertheimer

Contents

3 Installing Oracle Secure Backup on Windows

4 Oracle Secure Backup User Interfaces

5 Configuring and Managing the Administrative Domain

6 Managing Security for Backup Networks

A Oracle Secure Backup Directories and Files

B Oracle Secure Backup obparameters Installation Parameters

Preface

This Preface contains these topics:

Audience

This guide is intended for system administrators and database administrators who install the Oracle Secure Backup software. These administrators might also perform backup and restore operations. To use this document, you must be familiar with the operating system environment on which you plan to use Oracle Secure Backup. To perform Oracle database backup and restore operations, you should also be familiar with Recovery Manager concepts.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at `http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc`.

Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit `http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info` or visit `http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs` if you are hearing impaired.

Related Documents

For more information about backing up and restoring file systems with Oracle Secure Backup, see the following Oracle resources:

- *Oracle Secure Backup Reference*

  This manual contains information about the command-line interface for Oracle Secure Backup.

- *Oracle Secure Backup Administrator's Guide*

This book describes how to use Oracle Secure Backup to perform backup and restore operations. The book is oriented to the Oracle Secure Backup Web tool, which is a Web-based GUI interface.

For more information about database backup and recovery, including the Recovery Manager (RMAN) utility, see the following Oracle resources:

- *Oracle Database Backup and Recovery Advanced User's Guide*

  This book provides an overview of backup and recovery and discusses backup and recovery strategies. It provides instructions for basic backup and recovery of your database using Recovery Manager (RMAN).

The Oracle Secure Backup product site is located at the following URL:

`http://www.oracle.com/technetwork/products/secure-backup/overview/index.html`

You can download the Oracle Secure Backup software from the Download tab on this page.

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

1

Introduction to Oracle Secure Backup

This chapter provides an introduction to Oracle Secure Backup and includes advice on planning and configuring your **administrative domain**.

This chapter contains these sections:

- What Is Oracle Secure Backup?
- Oracle Secure Backup Concepts
- Oracle Secure Backup Interfaces
- System Requirements for Oracle Secure Backup
- Acquiring Oracle Secure Backup Installation Media
- Installation and Configuration Overview
- About Upgrade Installations

> **See Also:** *Oracle Secure Backup Administrator's Guide* for conceptual information about Oracle Secure Backup

What Is Oracle Secure Backup?

Oracle Secure Backup enables reliable data protection through **file-system backup** to tape. It supports every major **tape drive** and **tape library** in **SAN**, Gigabit Ethernet (GbE), and **SCSI** environments using standard tape formats.

Oracle Secure Backup supports Internet Protocol v4 (IPv4), Internet Protocol v6 (IPv6), and mixed IPv4/IPv6 environments on all platforms that support IPv6.

Using Oracle Secure Backup on your network enables you to take data from a networked host running Oracle Secure Backup or a **NAS** device that support **NDMP**, and back up that data on a **tape device** on the network. That data can include ordinary file-system files and databases backed up with **Recovery Manager (RMAN)**.

As part of the Oracle storage solution, Oracle Secure Backup provides scalable distributed backup and recovery capabilities. It reduces complexity of your backup solution, by:

- Integrating with the Oracle stack for maximum ease of use in a single Oracle solution to back up your data from disk to tape
- Employing single-vendor technical support for database and file-system backup and recovery to tape
- Using existing or new hardware, with broad tape device support in SCSI, GbE, and SAN environments with dynamic tape drive sharing for maximum tape drive utilization

Oracle Secure Backup eliminates integration challenges with ready-to-use tape management software that provides single-vendor support. Oracle Secure Backup also reduces your costs. When using Oracle Secure Backup with RMAN to back up and recover databases and files to and from tape, no third-party tape management software is required. Oracle Secure Backup provides the media management layer needed to use tape storage with RMAN.

Centralized administration, **heterogeneous network** support, and flexible scheduling simplify and automate protection of the entire Oracle environment, including database data and file-system data such as the contents of the Oracle home.

Oracle Secure Backup Concepts

This section discusses Oracle Secure Backup concepts that enable you to better understand the installation process.

This section contains these topics:

Oracle Secure Backup Administrative Domains and Hosts

Oracle Secure Backup organizes hosts and tape devices into an administrative domain, representing the network of hosts containing data to be backed up, hosts with attached tape devices on which backups are stored, and each **tape device** with its **attachment** to the hosts. A host can belong to only one administrative domain.

Host Roles in an Administrative Domain

Each host in an administrative domain must be assigned one or more of the following Oracle Secure Backup **roles**:

- **Administrative server**

  Each administrative domain must have exactly one **administrative server**. During postinstallation configuration, the administrative server must be configured with complete data regarding the other hosts in the administrative domain, their roles, and their attached tape devices. This configuration information is maintained in a set of configuration files stored on the administrative server.

  The administrative server runs the **scheduler**, which starts and monitors each **backup job**. The scheduler also keeps a backup **catalog** with metadata for all backup and restore operations performed in the administrative domain.

- **Media server**

  A **media server** is a host with at least one tape device attached to it. A media server transfers data to or from a **volume** loaded on one of these tape devices. A media server has at least one attachment to a tape drive or library. It might have attachments to multiple tape libraries.

  You specify the attachments between media servers and tape devices during postinstallation configuration of Oracle Secure Backup.

- **Client**

  The **client** role is assigned to any host that has access to file-system or database data that can be backed up or restored by Oracle Secure Backup. Any host where

Oracle Secure Backup is installed can be a client, including hosts that are also media servers or the administrative server. A network-attached **storage device** that Oracle Secure Backup accesses through NDMP can also serve the client role.

Note: A host can be assigned multiple roles in an administrative domain. For example, a host with a tape drive attached could be both the administrative server and media server for a network that includes several other clients. For more examples of administrative domains, see "Oracle Secure Backup Administrative Domain: Examples" on page 1-4.

See Also: "Choosing Secure Hosts for the Administrative and Media Servers" on page 6-6

Host Naming in an Administrative Domain

You must assign each host in an administrative domain a unique name to be used in Oracle Secure Backup operations. Typically, the host name in your DNS for this host is a good choice for the Oracle Secure Backup host name. However, you can assign a different name to a host.

Oracle Secure Backup Host Access Modes

Communication among hosts in an administrative domain is always based on NDMP, but implementations and versions of NDMP vary. Oracle Secure Backup supports two host access modes: **primary access mode** and **NDMP access mode**.

Primary access mode is used among hosts on which Oracle Secure Backup is installed. Oracle Secure Backup **daemons** run in the background on the host, communicate with the administrative server using the Oracle Secure Backup implementation of NDMP, and perform backup and restore tasks. Hosts on which databases reside are typically accessed using primary access mode.

Note: In Oracle Enterprise Manager, primary access mode is referred to as **native access mode**. In the Oracle Secure Backup Web tool and the output of some `obtool` commands such as `lshost`, primary mode is referred to as **OB access mode**.

NDMP access mode is used to communicate with devices such as storage appliances that do not run Oracle Secure Backup natively. For example, devices from third-party vendors such as Network Appliance and EMC are supported only in NDMP access mode. Each NDMP host uses a vendor-specific implementation of the NDMP protocol to back up and restore file systems. Some devices support older versions of the NDMP protocol. When adding such devices to the administrative domain, extra parameters might be required.

Oracle Secure Backup supports NDMP versions 3 and 4, and various extensions to version 4. It automatically negotiates with other, non-Oracle NDMP components to select a mutually supported protocol version. Between its own components, Oracle Secure Backup uses NDMP version 4. When communicating with hosts that are not running Oracle Secure Backup, Oracle Secure Backup usually chooses the protocol version proposed by that host when the connection is established. You can change the NDMP protocol version with which Oracle Secure Backup communicates to a specific host. You might want to do this when testing or troubleshooting.

Oracle Secure Backup Administrative Domain: Examples

Figure 1–1 shows a minimal administrative domain, in which a single host is administrative server, media server, and client. An Oracle database also runs on the same host.

Figure 1–1 Administrative Domain with One Host

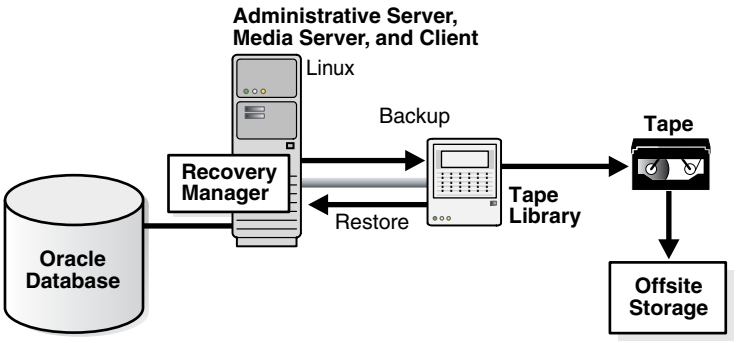
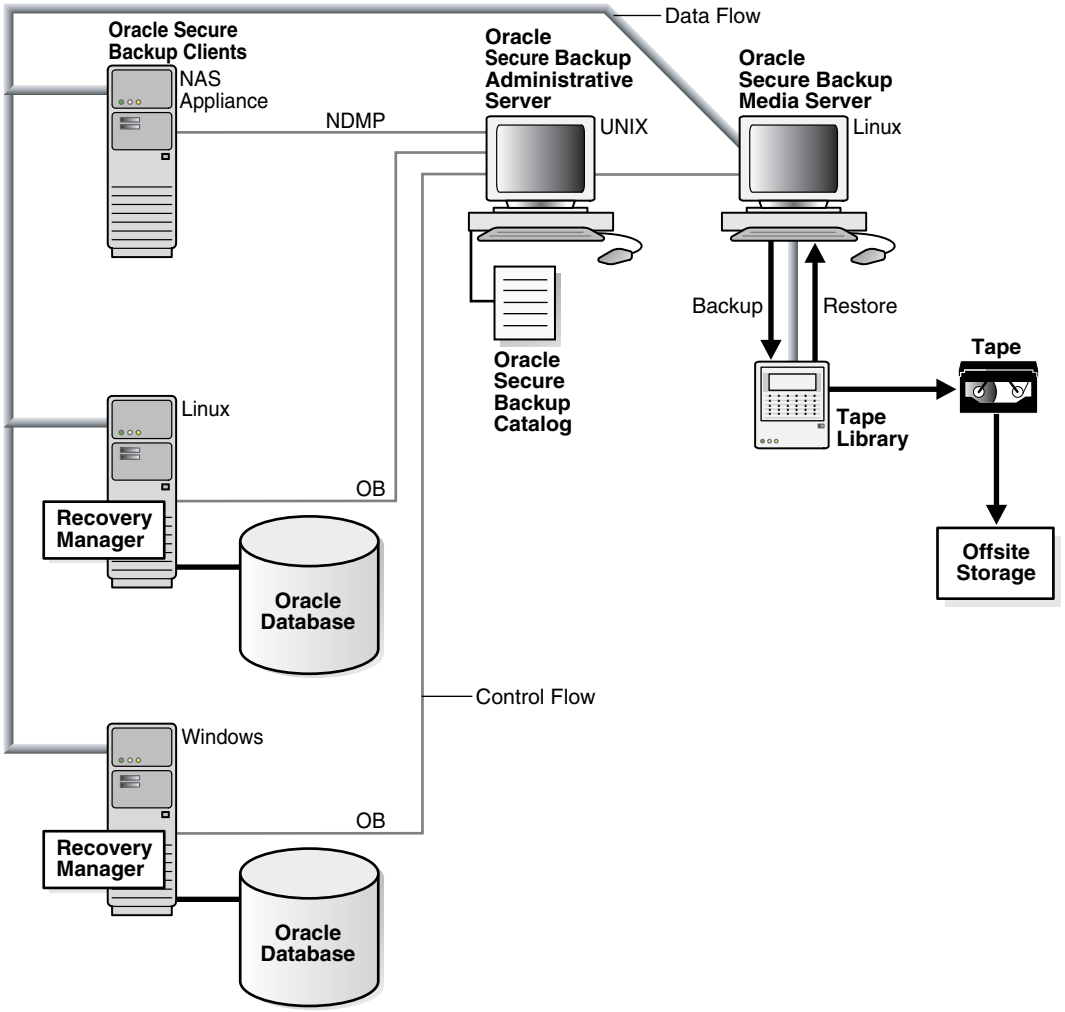


Figure 1–2 shows a possible Oracle Secure Backup administrative domain that includes three client hosts, one administrative server, and one media server. A NAS appliance contains ordinary file data. One client based on UNIX and another based on Windows contain databases and other file data. Oracle Secure Backup can back up to tape the non-database files on file systems accessible on client hosts. RMAN can back up to tape database files through the Oracle Secure Backup **SBT interface**.

Figure 1–2 Oracle Secure Backup Administrative Domain with Multiple Hosts



Tape Devices

Oracle Secure Backup maintains information about each tape library and tape drive so that you can use them for local and network backup and restore operations. You can configure tape devices during installation or add a new tape device to an existing administrative domain. When configuring tape devices, the basic task is to inform Oracle Secure Backup about the existence of a tape device and then specify which media server can communicate with this tape device.

This section contains these topics:

- Tape Drives
- Tape Libraries
- Device Names and Attachments

Tape Drives

A tape drive is a tape device that uses precisely controlled motors to wind a tape from one reel to another. The tape passes a read/write head as it winds. Most magnetic tape systems use small reels fixed inside a cartridge to protect the tape and make handling of the tape easier.

A magnetic cassette or tape is sequential-access storage. It has a beginning and an end, which means that to access data in the middle of the tape, a tape device must read through the beginning part of the tape until it locates the desired data.

In a typical format, a tape drive writes data to a tape in blocks. The tape drive writes each block in a single operation, leaving gaps between the blocks. The tape runs continuously during the write operation.

The **block size** of a block of data is the size of the block in bytes as it was written to tape. All blocks read or written during a given backup or restore operation have the same block size. The **blocking factor** of a block of data expresses the number of 512-byte records contained in the block. For example, the Oracle Secure Backup default blocking factor (128) results in a tape block size of 128*512 bytes or 64 KB.

The **maximum blocking factor** is an upper limit on the blocking factor that Oracle Secure Backup uses. This limit comes into play particularly during restores, when Oracle Secure Backup must pick an initial block size to use without knowing the actual block size on the tape. The maximum blocking factor limits this initial block size to a value that is acceptable to both the tape device and the underlying operating system.

When Oracle Secure Backup starts a backup, it decides what block size to use based on several factors. Listed in order of precedence, these factors are:

- Blocking factor specified using the `obtar -b` option

  This option can also be specified as part of the `operations/backupoptions` policy. If this option is specified, then it overrides all other factors.

  See Also: *Oracle Secure Backup Reference* for more information on the `obtar -b` option and the `operations/backupoptions` policy

- Configuration of the tape drive to be used

  You can specify what blocking factor, maximum blocking factor, or both that Oracle Secure Backup should use for a particular tape drive when you configure that drive. You might want to do this if you have tape drives with very different block size limits.

  See Also: "Configuring a Tape Drive" on page 5-17

- Domain-wide blocking factors or maximum blocking factors set with the `media/blockingfactor` and `media/maxblockingfactor` policies.

  See Also: *Oracle Secure Backup Reference* for more information on the `media/blockingfactor` and `media/maxblockingfactor` policies

- The default blocking factor (128) and maximum blocking factor (128), resulting in a block size of 64K

When a blocking factor has been nominated by one or another of these factors, it must pass the following tests:

- The block size must be less than or equal to the maximum block size (blocking factor) put in effect by whatever policies or tape drive configuration attributes are in force.
- The block size must be supported by the tape drive and attach point in question.

  Sometimes a tape drive, device driver, or kernel operating system has a limitation that supersedes all other considerations.

When Oracle Secure Backup begins a restore operation, it does not know what block size was used to write a given tape. Because issuing a read for a too-small block would result in an error condition and a tape reposition, Oracle Secure Backup always starts a restore operation by reading the largest possible block size. This is either the current setting of the `media/maxblockingfactor` policy or the tape drive configuration attribute. The maximum blocking factor, therefore, must always be greater than or equal to the largest block size you ever want to restore.

After the first read from the backup image, Oracle Secure Backup compares the amount of data requested to the actual size of the block and adjusts the size of subsequent reads to match what is on the tape.

Each tape drive supports a specific tape format. Typical tape formats include:

- 4mm, or Digital Audio Tape (DAT)
- Advanced Intelligent Tape (AIT)
- Digital Linear Tape (DLT) and Super DLT (SDLT)
- Linear Tape-Open (LTO)
- T9840
- T9940
- T10000

Information about the tape formats of tape devices supported by Oracle Secure Backup is available in the Getting Started section at the following URL:

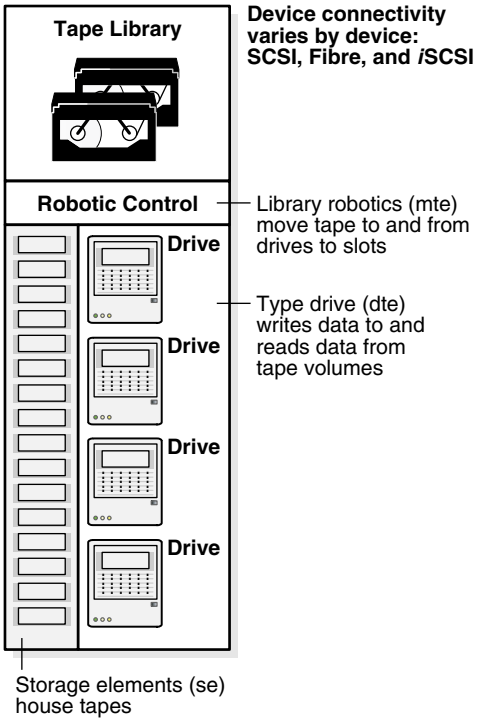
`http://www.oracle.com/technetwork/products/secure-backup/learnmore/index.html`

Tape Libraries

A tape library is a robotic tape device that accepts SCSI commands to move a volume between a **storage element** and a tape drive. A tape library is often referred to as a robotic tape device, autochanger, or medium changer.

A tape library contains one or more tape drives, slots to hold tape cartridges, and an automated method for loading tapes. Figure 1–3 illustrates a tape library that contains four tape drives.

Figure 1–3 Tape Library



Oracle Secure Backup automates the management of tape libraries, thereby enabling efficient and reliable use of their capabilities. Oracle Secure Backup controls the tape library robotics so that tapes can be managed easily.

Oracle Secure Backup supports the following features of tape libraries:

- Automatic loading and unloading of volumes

  When you add a tape library to your administrative domain, it is configured in automount mode by default. In this mode, Oracle Secure Backup sends commands to the robotic arm of the tape library to mount tapes for backup and restore operations. When a new volume is needed, Oracle Secure Backup scans the tape library until it finds a suitable volume. If sufficient eligible tapes are contained in the tape library storage elements, then no **operator** intervention is required to load the volumes needed to store the complete **backup image**.

- Barcode readers

  A **barcode** is a symbol code that is physically applied to volumes for identification purposes. Some tape libraries have an automated barcode reader. Oracle Secure Backup can use barcodes to identify tapes in a tape library.

- Automatic tape drive cleaning

  Oracle Secure Backup checks for cleaning requirements when a tape is loaded into or unloaded from a tape drive. If cleaning is required, then Oracle Secure Backup loads a cleaning cartridge, waits for the cleaning cycle to complete, replaces the cleaning cartridge in its original storage element, and continues with the requested load or unload. You can also schedule a cleaning interval.

As shown in Figure 1–3, a tape library has a set of addressable elements, each of which can contain or move a tape. Libraries can contain the following types of elements:

- Storage element (se)

This element is an internal slot in a tape library where a tape cartridge can reside.

- Data transfer element (dte)

  This element represents a tape device capable of reading or writing the physical volume. Typically, a **data transfer element (DTE)** is a tape drive used to back up or restore data on a tape.

- Medium transport element (mte)

  This element represents the robotics mechanism used to move tapes between other elements in the tape library. Typically, a medium transport element is a robot arm that moves tape cartridges from tape library slots to tape drives.

- Import/export element (iee)

  This is an element by which media can be imported to and exported from the tape library. Typically, an import/export element is a door-like mechanism that an operator uses to transfer tapes into and out of the library. After the door is closed, the robotic arm transfers cartridges to internal slots in the library. Because the library itself is not opened during this procedure, no re-inventory is required.

Many of the Oracle Secure Backup tape library commands require you to specify one or more tape library elements, in particular, storage elements and import/export elements. Except in the inventory display, media transport elements are never referenced. Data transfer elements are referenced only in the inventory display and indirectly by the tape drive (if any) that you select for an operation.

Oracle Secure Backup refers to elements by their abbreviation (`mte`, `se`, `iee`, or `dte`) followed by the number of the element, for example, `se5`, `iee2`, `dte1`. When multiple elements of a type exist, element numbering starts at 1. When only one element of a type exists, the number can be omitted. Thus, `iee1` and `iee` both refer to the first and only import/export element. If the abbreviation is omitted, then a storage element is assumed. For example, `se4` and `4` both refer to the fourth storage element. For some commands, you can specify a range of storage elements, for example, `1-5`.

Oracle Secure Backup supports several tape library operations. The following operations are the most basic:

- Inserting and extracting volumes
- Loading and unloading volumes
- Moving volumes
- Importing and exporting volumes

See Also:

- *Oracle Secure Backup Reference* for a description of the tape library commands that you can run in **obtool**

Virtual Tape Libraries

A virtual tape library is one or more large-capacity disk drives partitioned into virtual physical tape volumes. To Oracle Secure Backup the virtual tape library appears to be a physical tape library with at least one volume and at least one tape drive. The volumes and tape drives in the virtual tape library can be configured to match common physical tapes and tape drives.

Backup operations performed to a virtual tape library complete faster than backup operations to actual tape drives, because the underlying storage device is direct access media. But a virtual tape library is not suitable for long time storage, because it has

limited storage capacity. If you back up to a virtual tape library, then you can take advantage of its faster backup and then use the volume migration feature of Oracle Secure Backup to migrate the data to tapes at a later point of time.

Device Names and Attachments

Because Oracle Secure Backup manages tape drive operations, it must be able to identify the tape drive and determine whether the tape drive is housed in a tape library. Oracle Secure Backup must further determine if a storage element is available for storing a volume while not in use by the tape drive. Thus, each tape device must be uniquely identified within Oracle Secure Backup by a user-defined name.

Oracle Secure Backup distinguishes a tape device and the means by which the tape device connects to a host. To be usable by Oracle Secure Backup, each tape device must have at least one attachment, which describes a data path between a host and the tape device. An attachment usually includes the identity of a host plus an **attach point** name in Linux or UNIX, a device name in Windows, or a NAS device name. In rare cases, additional information is needed for the attachment definition.

See Also:

- "Adding Tape Devices to an Administrative Domain" on page 5-11 to learn how to configure a tape device
- *Oracle Secure Backup Reference* for a description of the `mkdev` command *aspec* placeholder, which describes the syntax and naming conventions for device attachments

Oracle Secure Backup Interfaces

There are four different interfaces for accessing different elements of Oracle Secure Backup:

- The **obtool** command line utility provides the fundamental interface for Oracle Secure Backup functions, including configuration, media handling, and backup and restore of file-system files.
- Oracle Enterprise Manager offers access to most Oracle Secure Backup functions available through `obtool` as part of its Database Control and Grid Control interfaces.
- Oracle Secure Backup includes its own Web-based interface, called the Oracle Secure Backup **Web tool**, which exposes all functions of `obtool`. The Oracle Secure Backup Web tool is primarily intended for use in situations where Oracle Secure Backup is being used independently of an Oracle Database instance. It does not provide access to database backup and recovery functions.

  The Oracle Secure Backup Web tool supports Internet Protocol v4 (IPv4), Internet Protocol v6 (IPv6), and mixed IPv4/IPv6 environments on all platforms that support IPv6.
- Backup and restore operations for Oracle Database instances and configuration of the Oracle Secure Backup media management layer are performed through the RMAN command-line client or through Oracle Enterprise Manager.

Note: Oracle Secure Backup documentation focuses on the use of Enterprise Manager wherever possible, and describes the Oracle Secure Backup Web Tool only when there is no equivalent functionality in Enterprise Manager, as in a **file-system backup**.

See also:

- Chapter 4, "Oracle Secure Backup User Interfaces" for details on using the different Oracle Secure Backup interfaces.
- *Oracle Database Backup and Recovery Advanced User's Guide* for details on using **Recovery Manager (RMAN)** for Oracle database backups

System Requirements for Oracle Secure Backup

For the list of operating systems, web browsers and **Network Attached Storage (NAS)** devices supported by Oracle Secure Backup, see Certify on My Oracle Support at the following URL:

`https://support.oracle.com`

Information about every **tape device** supported by Oracle Secure Backup is available at the following URL:

`http://www.oracle.com/technetwork/products/secure-backup/learnmore/index.html`

This section contains these topics:

- Disk Space Requirements for Oracle Secure Backup
- Other System Requirements for Oracle Secure Backup
- Linux Media Server System Requirement: SCSI Generic Driver

Disk Space Requirements for Oracle Secure Backup

When you install Oracle Secure Backup on Linux or UNIX, you load an install package for a particular operating system and perform the installation with the install package. Table 1–1 describes approximate disk space requirements.

Table 1–1 Disk Space Requirements for Oracle Secure Backup on Linux and UNIX

Oracle Secure Backup Installation	Disk Space
Linux	75 MB
Solaris	130 MB
HP-UX	130 MB
AIX	610 MB

Table 1–2 describes approximate disk space required for an installation of Oracle Secure Backup on Windows with and without the administrative server.

Table 1–2 Disk Space Requirements for Oracle Secure Backup on Windows

Oracle Secure Backup Installation	Disk Space
Administrative server (can include the media server, client, or both)	48 MB
Media server, client, or both (no administrative server)	31 MB

The disk space required for the Oracle Secure Backup **catalog** depends on many factors. But as a general rule, plan for catalog space equal to 250% of your largest index created after a backup.

See Also: *Oracle Secure Backup Administrator's Guide* for guidelines on the growth of the Oracle Secure Backup catalog over time

Other System Requirements for Oracle Secure Backup

Each host that participates in an Oracle Secure Backup **administrative domain** must run **TCP/IP**. Oracle Secure Backup uses this protocol for all communication within each of its components and between its components and other system components.

Each appliance that employs a closed operating system, such as **Network Attached Storage (NAS)** and tape servers, must support a version of **Network Data Management Protocol (NDMP)** described in "Oracle Secure Backup Host Access Modes" on page 1-3.

Each host that participates in an Oracle Secure Backup administrative domain must also have some preconfigured way to resolve a host name to an IP address. Most systems use DNS, NIS, WINS, or a local hosts file to do this. Oracle Secure Backup does not require a specific mechanism. Oracle Secure Backup only requires that, upon presenting the underlying system software with an IP address you have configured, it obtains an IP address corresponding to that name.

The use of DHCP to assign IP addresses is not supported for hosts that participate in an Oracle Secure Backup administrative domain. Static IP addresses should be assigned to all hosts. If you cannot use static IP addresses, then you must ensure that the DHCP server guarantees that a given host is always assigned the same IP address.

Note: You can change the static IP of a host from one address to another, but you must restart the Oracle Secure Backup **administrative server** for the change to take effect.

On Oracle Secure Backup network installations, it is important that there be no duplicate host names. Index catalog data is stored in a directory based on the name of the **client** host. Duplicate host names would result in information related to backups from multiple clients being combined in a manner that could prevent successful restore operations from backup files.

You can configure Oracle Secure Backup to use WINS, the Microsoft Windows name resolution protocol, from UNIX hosts. Although this configuration is atypical, WINS name resolution from UNIX hosts can be a practical solution.

Linux Media Server System Requirement: SCSI Generic Driver

Configuring a Linux host for the Oracle Secure Backup **media server** role requires that the SCSI Generic driver be installed on that host. This driver is required for Oracle Secure Backup to interact with a **tape device**. The host must also be configured to automatically reload the driver after a restart.

See Also: "Prerequisites for Installation on Linux" on page 2-2

Acquiring Oracle Secure Backup Installation Media

Oracle Secure Backup installation media for each supported platform is available as a CD-ROM or as a ZIP file downloaded from the Oracle Technology Network (OTN) Web site for Oracle Secure Backup:

`http://www.oracle.com/technetwork/products/secure-backup/downloads/index.html`

The contents of the CD-ROM and download archive are identical.

If you download the software from OTN, then you must store the downloaded file in a temporary directory and extract the contents of the installation file.

> **Note:** If you are installing Oracle Secure Backup on multiple platforms, then you must download the ZIP file or acquire the CD-ROM for each platform.

Installation and Configuration Overview

You must install Oracle Secure Backup on your **administrative server** and on each **media server** and **client** host in your **administrative domain**. During installation, the installation software asks you to specify the **roles** played by each host. An administrative domain typically includes an administrative server, one or more media servers, and one or more client hosts.

The following steps provide an overview of Oracle Secure Backup installation and configuration:

1. Create an Oracle Secure Backup administrative server.
   a. Select a host to be the administrative server. This is the host you use to initiate and manage backup and restore jobs.
   b. Verify that this host meets the physical and network security requirements discussed in "Choosing Secure Hosts for the Administrative and Media Servers" on page 6-6
   c. Verify that this host meets the system requirements discussed in "Disk Space Requirements for Oracle Secure Backup" on page 1-11.
   d. Install Oracle Secure Backup software on this host.

   When this step is complete, the administrative domain is initialized. But the only host included in the administrative domain at this point is the administrative server.
2. Create Oracle Secure Backup media servers.
   a. Select one or more hosts to be media servers. These hosts must have a **tape device** or other secondary **storage device** attached.
   b. Verify that this host meets the physical and network security requirements discussed in "Choosing Secure Hosts for the Administrative and Media Servers" on page 6-6
   c. Verify that this host meets the system requirements discussed in "Disk Space Requirements for Oracle Secure Backup" on page 1-11.
   d. Install Oracle Secure Backup software, including the Oracle Secure Backup device driver, on each of these hosts.

      On UNIX and Linux platforms you are prompted during this step for **Small Computer System Interface (SCSI)** device information. You obtain this information using operating system-specific utilities, as described in Appendix C, "Determining Linux SCSI Parameters".
3. Create Oracle Secure Backup clients.

   Install Oracle Secure Backup software on each host with data to be backed up.
4. Configure the Oracle Secure Backup administrative domain.

The administrative server requires complete information about:

- Each media server
- Each **tape device**
- Each **attachment** that associates a tape device with a media server
- Client hosts, including any **Network Data Management Protocol (NDMP)** clients such as **Network Attached Storage (NAS)** appliances

This step is documented in Chapter 5, "Configuring and Managing the Administrative Domain". When this step is complete, Oracle Secure Backup is ready to back up any data stored on clients in the administrative domain.

About Upgrade Installations

If you are upgrading an existing Oracle Secure Backup release 10.1 installation to release 10.4, then you must upgrade every host in the Oracle Secure Backup administrative domain to the same version. Oracle Secure Backup release 10.4 is incompatible with Oracle Secure Backup release 10.1.

In an upgrade installation, the Oracle Secure Backup catalogs (contained in the `admin` directory) are preserved, retaining configuration information and backup metadata for your administrative domain. This state information for your administrative domain, such as the backup catalog, host, user and device configuration information, and any scheduled backup jobs, is stored in the `admin` directory under the Oracle Secure Backup home on your administrative server.

Note: Oracle recommends backing up the administrative server before upgrading.

Before upgrading an existing administrative domain to Oracle Secure Backup release 10.4, you must shut down drivers and background processes related to Oracle Secure Backup on all hosts. Upgrade the administrative server host first, and then the other hosts in the domain.

Brief instructions on each step are described in the following sections.

Preparing Administrative Domain Hosts for Upgrade to Release 10.4

Before performing an upgrade installation, you must stop the daemons and services related to Oracle Secure Backup on all hosts in your administrative domain. The preferred commands for stopping the Oracle Secure Backup daemons on Linux and UNIX are described in *Oracle Secure Backup Reference*.

On both Linux and Solaris administrative servers, it is also necessary to stop the Oracle Secure Backup Web tool processes and Oracle Secure Backup httpd daemon processes. Use the `ps` command to confirm that all the Oracle Secure Backup processes are stopped:

```
# /bin/ps -ef | grep ob
```

Use the `kill -9` command to stop each process.

On Windows hosts, you must stop the Oracle Secure Backup service:

1. Open the Services applet.
2. Right-click the **Oracle Secure Backup Services** service.

3. Select **Stop**.

2

Installing Oracle Secure Backup on Linux or UNIX

This chapter explains how to install Oracle Secure Backup on hosts running Linux or UNIX.

This chapter contains the following sections:

Overview of Oracle Secure Backup Linux and UNIX Installation

There are three steps to installing Oracle Secure Backup on a Linux or UNIX host:

1. Loading

   Files required for installing Oracle Secure Backup are staged on the **administrative server**, in a directory called the **Oracle Secure Backup home**. This step is performed by a script named `setup`.

2. Installing

   Oracle Secure Backup executables are deployed correctly for use on the host. This step is performed by a script named `installob`.

Note: On a Solaris **media server**, `installob` also performs some **tape device** configuration tasks, including installation of a required device driver, and, optionally, **attach point** creation required for Oracle Secure Backup to access tape devices.

3. Creating attach points on each media server

   This step is required for the Oracle Secure Backup device driver to access tape devices. You need the **SCSI** device parameters to perform this task.

Note: If you are installing Oracle Secure Backup in an Oracle Real Application Clusters (Oracle RAC) environment, then you must install Oracle Secure Backup on each node in the cluster.

Prerequisites for Installing Oracle Secure Backup on Linux and UNIX

The prerequisites for installing Oracle Secure Backup on Linux and UNIX operating systems are:

- Each host must have a network connection with a static IP address and run **TCP/IP**.
- The `uncompress` utility must be installed on your system.

Note: If the uncompress utility is not installed on your system, then you can create an `uncompress` symbolic link pointing to the gunzip utility with the following command:

```
ln -s /bin/gunzip uncompress
```

- You must have the SCSI parameters for each **tape drive** and **tape library** attached to your Linux or UNIX media server. You can find them using the procedures in Appendix C, "Determining Linux SCSI Parameters". You need this information when creating an attach point for each tape device.
- On a Redhat Linux system, ensure that you install the `sg3_utils` and the `sg3_utils-libs` RPM packages. These packages are required for successfully running the `sg_map` command.
- You must be able to log in to each host with `root` privileges to perform the installation.

Prerequisites for Installation on Linux

For each Linux media server, ensure that the SCSI Generic (SG) driver is installed. This driver is required for Oracle Secure Backup to interact with a tape device.

Kernel modules are usually loaded directly by the facility that requires them, if the correct settings are present in the `/etc/modprobe.conf` file. However, it is sometimes necessary to explicitly force the loading of a module at start time.

For example, on RedHat Enterprise Linux, the module for the SCSI Generic driver is named `sg`. Red Hat Enterprise Linux checks at start time for the existence of the `/etc/rc.modules` file, which contains various commands to load modules.

Note: The `rc.modules` file is necessary, and not `rc.local`, because `rc.modules` runs earlier in the start process.

On RedHat Enterprise Linux, you can use the following commands to add the `sg` module to the list of modules configured to load as `root` at start time:

```
# echo modprobe sg >> /etc/rc.modules
# chmod +x /etc/rc.modules
```

An **Oracle Secure Backup user** must be mapped to a Linux or UNIX user that has read/write permissions to the `/dev/sg` devices. One way to accomplish this goal is to set the permissions to `666` for the `/dev/sg` devices.

Required SCSI Tape Device Parameters on Linux and UNIX

Oracle Secure Backup supports both SCSI and **Fibre Channel** devices for Linux and UNIX. To configure a media server to communicate with its attached tape devices, you must have the SCSI parameters for each tape device.

Table 2–1 lists the required SCSI parameters for each platform.

Table 2–1 Required SCSI Parameters

Platform	Linux	HP-UX	AIX
Host bus adapter	x	x	
SCSI bus address[1]	x	x	
SCSI bus name-instance	x	x	x
Target ID	x	x	x
SCSI LUN	x	x	x

[1] In Linux, SCSI bus addresses are referred to as channels.

You must also assign each tape drive and tape library an **Oracle Secure Backup logical unit number**, as described in "Assigning Oracle Secure Backup Logical Unit Numbers to Devices" on page 2-3.

Note: Do not confuse the SCSI LUN with the Oracle Secure Backup LUN. The SCSI LUN is part of the hardware address of the tape device, while the Oracle Secure Backup logical unit number is part of the device special file name.

Assigning Oracle Secure Backup Logical Unit Numbers to Devices

Each tape drive and tape library must be assigned an Oracle Secure Backup LUN during the configuration process. This number is used to generate unique device names during device configuration. Oracle Secure Backup logical unit numbers are assigned as needed automatically on Windows. For each UNIX or Linux media server, however, you must select Oracle Secure Backup logical unit numbers for each device as part of planning your administrative domain.

There is no required order for assigning Oracle Secure Backup logical unit numbers. They are typically assigned sequentially, starting at 0, for each tape device of a given type, whether tape library or tape drive. That is, tape libraries are typically numbered

0, 1, 2 and so on, and tape drives are also numbered 0, 1, 2 and so on. The maximum value for an Oracle Secure Backup logical unit number is 31.

On Linux or UNIX, the resulting device special file names for tape libraries are `/dev/obl1`, `/dev/obl2`, `/dev/obl3` and so on, and the names for tape drives are `/dev/obt1`, `/dev/obt2`, `/dev/obt3` and so on. On Windows, the resulting tape library names are `//./obl1`, `//./obl2`, `//./obl3` and so on, and the names for tape drives are `//./obt1`, `//./obt2`, `//./obt3` and so on, where these names are assigned automatically during the installation of Oracle Secure Backup on Windows.

See Also: "Identifying and Configuring Linux Attach Points" on page 2-18

Note: The Oracle Secure Backup logical unit number should not be confused with the SCSI LUN. The latter is part of the hardware address of the tape device, while the Oracle Secure Backup logical unit number is part of the device special file name.

Extracting Oracle Secure Backup from OTN Download on Linux or UNIX

This section explains how to download the Oracle Secure Backup software.

To download and extract the Oracle Secure Backup installation software:

1. Log in to your host as a user with `root` privileges.
2. Create a directory called `osbdownload` on a file system with enough free space to hold the downloaded installation file:

   ```
   mkdir /tmp/osbdownload
   ```

3. Open a Web browser and go to the Oracle Secure Backup Web site on Oracle Technology Network (OTN):

   `http://www.oracle.com//technetwork/products/secure-backup/overview/index.html`

4. Click the Downloads tab.

   The displayed page contains the following sections: Oracle Secure Backup Downloads and Oracle Secure Backup Express Downloads.

5. Click the version of Oracle Secure Backup that you want to download.

   The Oracle Secure Backup Downloads page is displayed.

6. Click **OTN License Agreement.**

   The Oracle Technology Network Developer License Terms page is displayed is a separate browser window.

7. Read the Export Controls on the Programs and close the window.

   You must accept the OTN License Agreement to download the Oracle Secure Backup software.

8. Select the **Accept License Agreement** option, and click the link for the version of Oracle Secure Backup release 10.4 specific to your operating system.

Note: If you have multiple operating systems in your environment, then you must perform multiple downloads of the Oracle Secure Backup release 10.4 software.

9. Save the Oracle Secure Backup release 10.4 installation software to a temporary directory.
10. Expand the compressed installation software to the `osbdownload` directory you created in step 2.

You now have all of the files required to install Oracle Secure Backup release 10.4.

Preparing to Install Oracle Secure Backup on Linux and UNIX

Perform the following actions before installing Oracle Secure Backup:

- Select hosts for the administrative server, media server, and client **roles**, as described in "Installation and Configuration Overview" on page 1-13.
- Collect the SCSI parameters for each tape drive and tape library attached to your Linux and UNIX media servers. You need this information when creating an attach point for each tape device.
- Disable any system software that scans and opens arbitrary SCSI targets before adding Oracle Secure Backup tape devices to an administrative domain. If Oracle Secure Backup must contend with other system software (such as monitoring software) for access to tape libraries and drives, then unexpected behavior can result.
- If you are installing Oracle Secure Backup in an Oracle RAC environment, then you must install Oracle Secure Backup on each node in the cluster.

Creating the Oracle Secure Backup Home

You must create an Oracle Secure Backup home. The Oracle Secure Backup `setup` program uses this directory to store installation files specific to your host.

Note: Oracle recommends that you use `/usr/local/oracle/backup` as your Oracle Secure Backup home. If you use a different directory, then the setup program prompts you to confirm your selected directory.

Note: To enable users other than `root` to use `obtool` or the Oracle Secure Backup Web tool, install Oracle Secure Backup to a file system that can use the `suid` mechanism. You can do this by excluding the `nosuid` option from the `/etc/fstab` file entry for that file system.

See also: "Oracle Secure Backup Home Directory" on page A-1 and *Oracle Secure Backup Administrator's Guide* for more details about the Oracle Secure Backup home.

To create the Oracle Secure Backup home:

1. Log into the host as `root`.
2. Run the following command:

```
# mkdir -p /usr/local/oracle/backup
```

Loading Oracle Secure Backup Software on Linux or UNIX Using setup Script

The setup script performs the loading process, in which packages of files required to install Oracle Secure Backup are extracted from the installation media and staged in the Oracle Secure Backup home for later use by the `installob` installation script.

To load Oracle Secure Backup into an Oracle Secure Backup home directory for later installation on one or more Linux or UNIX platforms:

1. Log into your Linux or UNIX operating system as `root`.
2. Change to the Oracle Secure Backup home directory created in "Creating the Oracle Secure Backup Home" on page 2-5. For example:

```
# cd /usr/local/oracle/backup
```

3. Run the `setup` script from your installation media or extracted archive directory. Enter the following command, where *`/media_dir`* is the CD-ROM mount point or the directory containing the files extracted from the downloaded archive:

```
# /media_dir/setup
```

   For example, if you downloaded an archive from Oracle Technology Network (OTN) and extracted the setup software to the `/tmp/osbdownload/OB` directory, then you would run `setup` as follows:

```
# /tmp/osbdownload/OB/setup
```

   Oracle Secure Backup expands compressed files in a temporary directory during installation. To specify a directory for this expansion, you can use the `-t` option to the `setup` command. The following example specifies that `setup` should use *`directory_name`* for the expansion:

```
# /media_dir/setup -t directory_name
```

   The `setup` script displays the following messages:

   - A welcome message stating the Oracle Secure Backup version number and then displays progress messages
   - A message stating the platform
   - Various progress messages as it loads the package

   When the script finishes, it prompts you to unmount and remove the installation CD-ROM.

   Note: At this point the loading process is complete. The files required to install Oracle Secure Backup are stored in the Oracle Secure Backup home on this host.

4. The setup script prompts you to start the `installob` script to install Oracle Secure Backup on the local host. Choose one of these options:

- Enter `no` to run `installob` later, or if you must customize some aspect of your installation process using the `obparameters` file, as described in "Configuring Installation Parameters in the obparameters File" on page 2-7.

  If you enter no, then `setup` tells you how to continue installation later, and setup exits.

  See Also: "Installing Oracle Secure Backup on Linux or UNIX with installob" on page 2-8 for instructions on starting `installob`

- Enter `yes` to start the `installob` script. The steps for running `installob` are described in "Installing Oracle Secure Backup on Linux or UNIX with installob" on page 2-8.

Note: If the setup script is interrupted, then some temporary files, named `OB`*`nnnn`* or `OB`*`nnnn`*`.Z`, might remain in `/usr/tmp`. You can safely delete these files.

Configuring Installation Parameters in the obparameters File

The setup script creates a file called `obparameters` in the install subdirectory of the Oracle Secure Backup home. For example, if the Oracle Secure Backup home is in the default location `/usr/local/oracle/backup`, then the `obparameters` file is located at `/usr/local/oracle/backup/install/`.

During the installation process the setup script gives you the choice of accepting the default settings in the `obparameters` file or customizing those settings. In most cases, it is not necessary to change the defaults in the `obparameters` file. However, you should review the parameters you can control in this file as part of planning your installation, and determine whether any of them should be changed.

The `obparameters` file is plain text that can be edited using any standard text editor.

Reasons to change the parameters in the `obparameters` file include:

- You can specify a different key size for enhanced security or performance

  See Also: "Setting the Key Size in obparameters" on page 6-19

- You can customize installation directories and symbolic links created during installation on different platforms.
- If you are using Oracle Secure Backup to back up Oracle Database files to tape, then you can create an Oracle Secure Backup user named `oracle` for use in RMAN backups. You can associate this user with Linux or UNIX operating system credentials by setting parameters in `obparameters`.

Note:

- You can also configure a preauthorized `oracle` user later. Before electing to create an Oracle Secure Backup `oracle` user, be aware that this choice involves a trade-off between convenience and security.
- If you intend to use Oracle Secure Backup to perform one-time, RMAN-initiated, or **unprivileged backup** operations on Windows clients, then you must modify the Oracle Secure Backup `admin` and `oracle` users to assign them Windows credentials (a domain, user name and password) that are valid at the **client** with required privileges after you complete the Oracle Secure Backup installation. Otherwise, Oracle Secure Backup cannot perform these types of backup operations. This requirement applies regardless of the platform that acts as the administrative server.

See Also:

- *Oracle Secure Backup Administrator's Guide* for more information about the preauthorized `oracle` user and RMAN backups.
- Appendix B, "Oracle Secure Backup obparameters Installation Parameters"

Installing Oracle Secure Backup on Linux or UNIX with installob

To install the Oracle Secure Backup software on Linux or UNIX:

1. Ensure that the SCSI parameters for each tape device available.

   You can enter these parameters to create an attach point for each SCSI device as part of the initial installation. Solaris 10 systems have special device configuration procedures. See "Configuring the Solaris sgen Driver to Provide Oracle Secure Backup Attach Points" on page 2-19.

2. Start the `installob` script.

   The Oracle Secure Backup setup script ends by asking to start the installation process using the `installob` script. If you enter `yes` to this question, then the setup script runs the `installob` script for you.

   Otherwise, start `installob` from the command prompt. While logged in as `root`, go to the Oracle Secure Backup home and enter the following command:

   ```
   install/installob
   ```

   The `installob` script displays a welcome message and tells you that most of its questions have default answers, which you can select by simply pressing **Enter**.

3. Confirm the settings in the `obparameters` file.

   This step depends upon the value of the `customized obparameters` parameter in the `obparameters` file described in "Configuring Installation Parameters in the obparameters File" on page 2-7. The two possibilities are:

   - You have edited the `obparameters` file and set `customized obparameters` to `yes`.

In this case, the `installob` script assumes that you have made the changes you want in the `obparameters` file and uses those parameters during the installation. Continue to step 4.

- The `customized obparameters` parameter is set to `no`, which is the default.

  In this case, the `installob` script asks if you have reviewed and customized the `obparameters` file. Choose one of these options:

  – Enter `yes` or press the Enter key to indicate that you do not want to customize the `obparameters` file. Continue to step 4.

  – Enter `no` to indicate that you do want to customize the `obparameters` file. The `installob` script tells you to rerun the script after reviewing `obparameters`. The `installob` script then exits.

  See Also: "customized obparameters" on page B-1 for details about the `customize obparameters` parameter.

4. Specify the host role.

   You determined the **roles** for each host when planning your administrative domain. Choose one of these options:

   - Enter **a** to install the software for an administrative server.

     If you choose this option, then `installob` also installs the software required for the media server and client roles.

   - Enter **b** to install the software for a media server.

     If you choose this option, then `installob` also installs the software required for the client role.

   - Enter **c** to install the software for a client.

   You can add or remove a role later with the `chhost` command in **obtool**.

   Note:

   - If you choose an administrative server or media server installation, then `installob` installs the *software* necessary for the media server role. However, the host does not have the media server *role* until the `admin` user grants that role with the `chhost` command after Oracle Secure Backup is installed.
   - To add the media server role to an administrative server or client after initial installation, you must create attach points using makedev or `installob`. See *Oracle Secure Backup Reference* for details.

   See Also: "Installation and Configuration Overview" on page 1-13 to learn more about the roles of administrative server, media server and client in Oracle Secure Backup

   This procedure describes installation for an administrative server.

5. Create a password for the Oracle Secure Backup keystore.

   The `installob` script prompts for a password for the keystore and then prompts you to re-enter the password. Oracle recommends that you choose a password of

at least 8 characters in length that contains a mixture of alphabetic and numeric characters. When you enter the password, the password is not echoed to the display.

6. Create a password for the Oracle Secure Backup administrative server.

   The `installob` script asks for a password for the `admin` user, and then asks you to reenter it for confirmation. Oracle recommends that you choose a password of at least 8 characters in length, containing a mixture of alphabetic and numeric characters. When you type in the password, your entry is not echoed to the display.

   The minimum password length is determined by the `minuserpasswordlen` security policy. Its default value is 0, which means a null password is permitted. You can change the value of `minuserpasswordlen` by setting the `minimum user password length` parameter in the `obparameters` file.

   See Also: *Oracle Secure Backup Reference* for more information on the minuserpasswordlen security policy

   Note: The practice of supplying a password in clear text on a command line or in a command script is not recommended by Oracle. It is a security vulnerability. The recommended procedure is to have the user be prompted for the password.

7. Enter an e-mail address for notifications.

   The `installob` script asks for an e-mail address to which Oracle Secure Backup sends notifications.

   Note: The default *from* address for e-mails generated by Oracle Secure Backup is `root@`*`fqdn`*, where *`fqdn`* is the fully qualified domain name of the Oracle Secure Backup administrative server. You can change this default *from* address after installation. See *Oracle Secure Backup Reference* for more information.

   The `installob` script now displays informational messages as it installs and configures the Oracle Secure Backup software on this host. This process might take several minutes to complete.

8. If you are installing Oracle Secure Backup on an administrative server or media server, then the `installob` script asks to configure a tape drive or tape library.

   Note: In `installob`, the term *configuring* refers to creating the attach points required for Oracle Secure Backup to communicate with the tape devices. Do not confuse this step with configuring the administrative domain with information about tape devices and media servers, as described in Chapter 5, "Configuring and Managing the Administrative Domain".

   The `installob` script includes software required for both the administrative server and media server roles in an administrative server installation. Therefore, this prompt is displayed when installing on an administrative server even if there are no attached tape drives or tape libraries.

Although this procedure discusses SCSI tape libraries and tape drives, it also applies to a Fibre Channel tape device.

Choose from these options:

- Enter `no` if you do not want to create attach points for your tape devices now, or if you are installing on an administrative server with no tape devices attached.

  Note: On Linux and Solaris systems Oracle recommends that you enter `no` when asked to configure tape libraries or drives during installation.

  On Linux, the recommended method is to use the `/dev/sg` devices for attach points, as described in "Identifying and Configuring Linux Attach Points" on page 2-18. For Solaris systems, see "Configuring the Solaris sgen Driver to Provide Oracle Secure Backup Attach Points" on page 2-19.

  If you choose to create attach points later, or if you add a tape device to a media server in the future, then see "Creating Attach Points" on page 2-13 for two alternative methods of completing this task.

- Enter `yes` to configure tape devices now.

  To create attach points, the `installob` script asks if tape libraries are connected to this host, and if so, what the SCSI parameters are for each tape library. After you have entered the tape library SCSI parameters, the `installob` script asks you to confirm your entries.

  When you have entered information about tape libraries attached to this host, the `installob` script asks the same questions about standalone tape drives.

  Table 2–2 lists the information required by `installob` for each platform. For the device type, enter a `d` for a tape drive or `l` (lowercase `L`) for a tape library.

Table 2–2 Information Required by installob

Platform	Linux	HP-UX	Solaris	AIX
Oracle Secure Backup LUN[1]	x	x	x	x
Device type	x	x	x	x
Host bus adapter	x	x		
SCSI bus address[2]	x	x		
SCSI bus name-instance	x	x	x	x
Target ID	x	x	x	x
SCSI LUN	x	x	x	x

[1] Do not confuse the Oracle Secure Backup logical unit number with the SCSI LUN.
[2] In Linux, SCSI bus addresses are referred to as channels.

Enter each parameter value in response to the prompts from the `installob` script. You can press Enter to accept a default value, but the default SCSI parameters offered by the script might not be correct.

When you have entered the SCSI parameters for all tape libraries and tape drives attached to this host, the `installob` script begins device driver configuration and device special file creation.

Record the name of the device special file created for each tape device. The file name is needed when you configure the **attachment** for the tape device, as part of configuring the Oracle Secure Backup domain. The file name should be `/dev/obt`*n* for tape drives, and `/dev/obln` for tape libraries, where *n* is the Oracle Secure Backup LUN you entered for the tape device.

If you enter the wrong parameters, then device special file creation fails. To resolve the resulting errors, run `installob` again, entering the correct values, or use the `makedev` script described in "Creating Attach Points" on page 2-13.

When the `installob` script has created attach points for all tape devices attached to this host, it reminds you that you must configure these tape devices through the Oracle Secure Backup Web interface or the command line using the `mkdev` command in `obtool`.

9. The `installob` script displays a summary of installation activities during this session and exits. This installation summary does not include any information about device special file creation performed during the `installob` session.

Installing or Uninstalling Oracle Secure Backup on AIX

The installation and uninstallation procedures for AIX and Linux/UNIX are identical.

During Oracle Secure Backup installation, the Oracle Secure Backup `admin` user is mapped by default to UNIX user `root` and UNIX group `root`. In this configuration, Oracle Secure Backup requires that the user `root` be a member of the group `root` to back up the file system successfully. AIX does not define a group `root` by default. If the group `root` does not exist on your AIX system, then you must create it and make user `root` a member of it.

Note: You can change this mapping of the Oracle Secure Backup `admin` after installation.

See Also:

- "Installing Oracle Secure Backup on Linux or UNIX with installob" on page 2-8 and "Uninstalling Oracle Secure Backup on Linux or UNIX" on page 2-21
- "Identifying and Configuring AIX Devices" on page 2-13

Installing or Uninstalling Oracle Secure Backup on HP-UX

The installation and uninstallation procedures for HP-UX and Linux/UNIX are identical.

See Also:

- "Installing Oracle Secure Backup on Linux or UNIX with installob" on page 2-8 and "Uninstalling Oracle Secure Backup on Linux or UNIX" on page 2-21
- "Identifying and Configuring HP-UX Devices" on page 2-16

Creating Attach Points

The `makedev` script in Oracle Secure Backup is used to create an attach point for a single tape device. Internally, the `installob` script calls `makedev` once for each tape device specified during installation. Alternatively, you can run `makedev` outside of `installob` to create all required attach points.

The `makedev` script can also replace an old attach point, rather than creating a new one. If you reuse an Oracle Secure Backup LUN for a tape library or drive, then the attach point for the old tape device is overwritten.

Table 2–3 lists the information required by makedev for each platform. For the device type, enter a `d` for a tape drive or `l` (lowercase `L`) for a tape library.

Table 2–3 Information Required by makedev

Platform	Linux	HP-UX	AIX
Oracle Secure Backup LUN[1]	x	x	x
Device type	x	x	x
Host bus adapter	x	x	
SCSI bus address	x	x	
SCSI bus name-instance	x	x	x
Target ID	x	x	x
SCSI LUN	x	x	x

[1] Do not confuse the Oracle Secure Backup logical unit number with the SCSI LUN.

See Also: *Oracle Secure Backup Reference* for `makedev` syntax

This section contains the following topics:

- "Identifying and Configuring AIX Devices" on page 2-13
- "Identifying and Configuring HP-UX Devices" on page 2-16
- "Identifying and Configuring Linux Attach Points" on page 2-18
- "Configuring the Solaris sgen Driver to Provide Oracle Secure Backup Attach Points" on page 2-19

Identifying and Configuring AIX Devices

To access SCSI or Fibre Channel tape devices, Oracle Secure Backup requires the following identifying information about how the devices are attached to their hosts:

- SCSI bus name
- Target ID
- LUN

This information may not be readily available for all attached devices using standard operating system commands.

Identifying and Configuring AIX Devices in a Switched Fibre Channel Configuration or Direct Attached SCSI Device Configuration

If you use Fibre Channel tape and media changer devices in a switched environment on AIX, you can use the standalone tool `obscan` to assist with gathering device information. The SCSI ID and LUN are required to correctly configure the devices for use by Oracle Secure Backup.

The `obscan` tool is provided as an optional tool for device identification in AIX environments. The `obscan` executable is located in the `cdtools` directory of the Oracle Secure Backup CD or CD image. The syntax is as follows, where *`dname`* is the device file name of the SCSI bus or Fibre Channel fabric to scan:

```
obscan dname
```

The `obscan` tool determines the SCSI ID and LUN for every tape and media changer device.

To identify and configure AIX devices with obscan and makedev:

1. Log on as `root`.

   You must have operating system privileges to access devices, which is often root access, to run `obscan`.

2. Run `obscan` for each SCSI and Fibre Channel adapter with tape devices to be used by Oracle Secure Backup.

   In the following example, `obscan` gathers information about the tape devices connected to the SCSI bus identified by the device file `/dev/scsi2`:

   ```
   obscan /dev/scsi2

   obscan version 10.4.0.3 (AIX)
   Copyright (c) 1992, 2012, Oracle. All rights reserved.

   DEVICE information for /dev/scsi2

    Target-id : 0, Lun : 0
       Vendor : ADIC  Product : FastStor 2

    Target-id : 5, Lun : 0
       Vendor : HP    Product : Ultrium 2-SCSI

    Total count of Media Changers and/or Tape devices found : 2
   ```

   In this second example, `obscan` gathers information about the tape devices connected to the Fibre Channel fabric identified by `/dev/fssci0`:

   ```
   obscan /dev/fscsi0

   DEVICE information for /dev/fscsi0

    Target-id : 6423827, Lun : 0
       Vendor : ADIC  Product : Scalar 24    World Wide Name : 2001006045175222

    Target-id : 6423827, Lun : 1
       Vendor : IBM   Product : ULTRIUM-TD2  World Wide Name : 2001006045175222

    Target-id : 6423827, Lun : 2
       Vendor : IBM   Product : ULTRIUM-TD2  World Wide Name : 2001006045175222

    Target-id : 6491411, Lun : 0
   ```

```
   Vendor : ADIC  Product : Scalar i500  World Wide Name : 2400005084800672

 Target-id : 6491411, Lun : 1
   Vendor : IBM   Product : ULTRIUM-TD3  World Wide Name : 2400005084800672

 Target-id : 6491411, Lun : 2
   Vendor : IBM   Product : ULTRIUM-TD3  World Wide Name : 2400005084800672

 Target-id : 6491411, Lun : 3
   Vendor : IBM   Product : ULTRIUM-TD3  World Wide Name : 2400005084800672

 Target-id : 6491411, Lun : 4
   Vendor : IBM   Product : ULTRIUM-TD3  World Wide Name : 2400005084800672

 Total count of Media Changers and/or Tape devices found : 8
```

3. Navigate to the `install` directory in your Oracle Secure Backup home. For example:

   ```
   # cd /usr/local/oracle/backup/install
   ```

4. Enter the `makedev` command at the shell prompt:

   ```
   # makedev
   ```

5. At the prompts, enter the information required to create attach points used within Oracle Secure Backup to identify devices for backup and restore operations.

   In the following example, the attach point `/dev/obl8` is created for the ADIC FastStor 2 library attached to `scsi2` having the target id `0` and lun `0`:

   ```
   makedev
   Enter logical unit number 0-31 [0]: 8
   Enter 'd' if this device is a tape drive or 'l' if a SCSI-2 addressable
       tape library [d]: l
   Enter SCSI bus name: scsi2
   Enter SCSI target id 0-16777215: 0
   Enter SCSI logica l unit number (lun) 0-7 [0]: 0
   /dev/obt8 created
   ```

   In this second example, the attach point `/dev/obl9` is created for the ADIC Scalar 24 library attached to `fsci0` having the target id `6423827` and lun `0`:

   ```
   makedev
   Enter logical unit number 0-31 [0]: 9
   Enter 'd' if this device is a tape drive or 'l' if a SCSI-2 addressable
       tape library [d]: l
   Enter SCSI bus name: fscsi0
   Enter SCSI target id 0-16777215: 6423827
   Enter SCSI logical unit number (lun) 0-7 [0]: 0
   /dev/obl9 created
   ```

   The `makedev` script creates the attach point, displaying messages indicating its progress.

Identifying and Configuring AIX Devices in a Point-to-Point or FC-AL Configuration

In a point-to-point or FC-AL configuration, no tool is provided to help you determine the SCSI ID and LUN. However, for IBM-supported devices in these configurations, you can use the `lsattr` command.

To identify and configure AIX devices with lsattr and makedev:

1. Log on as `root`.

   You must have operating system privileges to access devices, which is often root access, to run `lsattr`.

2. Run `lsattr` for each SCSI and Fibre Channel adapter with tape devices to be used by Oracle Secure Backup.

   The following `lsattr` example displays the attribute names, current values, descriptions, and user-settable flag values for the `rmt0` device:

```
user: lsattr -El rmt0
block_size     512                BLOCK size (0=variable length)           True
delay          45                 Set delay after a FAILED command         True
density_set_1  0                  DENSITY setting #1                       True
density_set_2  0                  DENSITY setting #2                       True
extfm          yes                Use EXTENDED file marks                  True
location                          Location Label                           True
lun_id         0x1000000000000    Logical Unit Number ID                  False
mode           yes                Use DEVICE BUFFERS during writes         True
node_name      0x1000006045175222 FC Node Name                            False
res_support    no                 RESERVE/RELEASE support                  True
ret_error      no                 RETURN error on tape change or reset     True
rwtimeout      144                Set timeout for the READ or WRITE commandTrue
scsi_id        0x2                SCSI ID                                 False
var_block_size 0                  BLOCK SIZE for variable length support   True
ww_name        0x2001006045175222 FC World Wide Name                      False
```

   You can convert the hexadecimal values of `lun_id` and `scsi_id` (shown in bold) to decimal so that they are usable by the Oracle Secure Backup `makdev` command. After conversion, the SCSI LUN ID is 281474976710656 and the SCSI ID is 2.

3. Navigate to the `install` directory in your Oracle Secure Backup home. For example:

```
# cd /usr/local/oracle/backup/install
```

4. Enter the `makedev` command at the shell prompt:

```
# makedev
```

5. At the prompts, enter the information required to create attach points used within Oracle Secure Backup to identify devices for backup and restore operations.

   The `makedev` script creates the attach point, displaying messages indicating its progress.

Identifying and Configuring HP-UX Devices

To access SCSI or Fibre Channel tape devices on HP-UX using the `makedev` script, Oracle Secure Backup requires the following identifying information about how the devices are attached to their hosts:

- SCSI bus number instance
- Target ID
- LUN

To gather device information in HP-UX, you can use the `ioscan` utility located in `/usr/sbin` on the HP-UX operating system. The `ioscan` command searches the system and lists any devices that it finds. You must have root access to run `ioscan`.

Note: The `ioscan` tool is provided as an optional tool for device identification in HP-UX environments. The `ioscan` tool is not included as part of any Oracle Secure Backup installation.

To identify and configure HP-UX devices:

1. Log on as `root`.
2. Execute the following command:

   ```
   /usr/sbin/ioscan -f
   ```

   Running the command with the `-f` option displays full information about the system configuration including device class, instance number, device or interface driver, software state, and hardware type.

   Example 2–1 shows sample output for `ioscan -f`. The bus number instance, target ID, SCSI LUN, and device description for each device are shown in bold.

Example 2–1 ioscan -f

```
$ /usr/sbin/ioscan -f

Class      I  H/W Path                Driver   S/W State  H/W Type   Description
...
ext_bus    3  0/1/1/1                 mpt      CLAIMED    INTERFACE  SCSI Ultra320
target    11  0/1/1/1.1               tgt      CLAIMED    DEVICE
autoch     4  0/1/1/1.1.0             schgr    CLAIMED    DEVICE     ADIC FastStor 2
target    10  0/1/1/1.2               tgt      CLAIMED    DEVICE
tape       8  0/1/1/1.2.0             stape    CLAIMED    DEVICE     HP  Ultrium 2-SCSI
...
fcp        2  0/2/1/0.99              fcp      CLAIMED    INTERFACE  FCP Domain
ext_bus    9  0/2/1/0.99.15.255.1     fcpdev   CLAIMED    INTERFACE  FCP Device Interface
target     1  0/2/1/0.99.15.255.1.3   tgt      CLAIMED    DEVICE
autoch     8  0/2/1/0.99.15.255.1.3.0 schgr    CLAIMED    DEVICE     ADIC Scalar 24
tape      19  0/2/1/0.99.15.255.1.3.1 stape    CLAIMED    DEVICE     IBM ULTRIUM-TD3
tape      20  0/2/1/0.99.15.255.1.3.2 stape    CLAIMED    DEVICE     IBM ULTRIUM-TD3
```

3. Using the `ioscan` output, make a note of the bus number, target ID, and SCSI LUN for the tape devices.

   Table 2–4 shows the relevant information from Example 2–1.

Table 2–4 Information Required by makedev

Device	Type	Name	Bus Number Instance	Target ID	SCSI LUN
Tape library (autoch)	SCSI	ADIC FastStor 2	3	1	0
Tape drive (tape)	SCSI	HP Ultrium 2	3	2	0
Tape library (autoch)	FC	ADIC Scalar 24	9	3	0
Tape drive (tape)	FC	IBM ULTRIUM-TD3	9	3	1
Tape drive (tape)	FC	IBM ULTRIUM-TD3	9	3	2

4. Use `makedev` to create attach points so that Oracle Secure Backup can identify devices for backup and restore operations.

The following example runs `makedev` using the information in Table 2–4. The example creates the attach point `/dev/obl/8` for the ADIC FastStor 2 library on SCSI bus instance 3 with the target ID 1 and SCSI LUN 0.

```
% makedev
Enter logical unit number 0-31 [0]: 8
Enter 'd' if this device is a tape drive or 'l' if a SCSI-2 addressable
    tape library [d]: l
Enter SCSI bus instance: 3
Enter SCSI target id 0-16777215: 1
Enter SCSI logical unit number (lun) 0-7 [0]: 0
/dev/obl/8 created
```

The following example runs `makedev` using the information in Table 2–4. The example creates the attach point `/dev/obt/9m` for the HP Ultrium 2 tape drive on SCSI bus instance 3 with the target ID 2 and SCSI LUN 0.

```
% makedev
Enter logical unit number 0-31 [0]: 9
Enter 'd' if this device is a tape drive or 'l' if a SCSI-2 addressable
    tape library [d]: d
Enter SCSI bus instance: 3
Enter SCSI target id 0-16777215: 2
Enter SCSI logical unit number (lun) 0-7 [0]: 0
/dev/obt/9m created
```

Identifying and Configuring Linux Attach Points

Oracle recommends that you use the `/dev/sg` devices as attach points with Oracle Secure Backup on Linux. The use of the Oracle Secure Backup `/dev/ob` devices has certain limitations that may not be acceptable in some environments. For example, the LUN cannot be greater than 7, and the SCSI bus number cannot be greater than 1. The existing method of using `/dev/ob*` devices continues to work for a tape device that does not fall into the limitation category.

To identify the `/dev/sg` that corresponds to the tape device you are interested in, you can use the `sg_map` command.

To configure Linux attach points:

1. Execute the following Linux command:

   ```
   sg_map -i -x
   ```

   Example 2–2 shows sample output.

Example 2–2 sg_map -i -x

```
sg_map -i -x
/dev/sg0   0 0 0 0  0  /dev/sda  DELL      PERC Stripe       V1.0
/dev/sg1   0 0 1 0  0  /dev/sdb  DELL      PERC Stripe       V1.0
/dev/sg2   0 0 2 0  0  /dev/sdc  DELL      PERC Volume       V1.0
/dev/sg3   1 0 1 0  8  ADIC      FastStor 2        G12r
/dev/sg4   1 0 2 0  1  /dev/nst0  HP        Ultrium 2-SCSI    F53A
/dev/sg5   2 0 0 0  1  /dev/nst1  IBM       ULTRIUM-TD2       5AT0
/dev/sg6   2 0 0 1  8  ADIC      Scalar 24         310A
/dev/sg7   2 0 1 0  1  /dev/nst2  IBM       ULTRIUM-TD2       5AT0
/dev/sg8   2 0 1 1  8  ADIC      Scalar 24         310A
/dev/sg9   2 0 2 0  1  /dev/nst3  IBM       ULTRIUM-TD3       54K1
/dev/sg10  2 0 3 0  1  /dev/nst4  IBM       ULTRIUM-TD3       54K1
/dev/sg11  2 0 3 1  8  ADIC      Scalar 24         310A
```

2. Using the `sg_map` output, make a note of the attach point for each tape device that you want to configure.

   Table 2–5 shows a tape library and tape drive from Example 2–2.

Table 2–5 Information Required by mkdev

Device Type	Name	Path
Tape library	ADIC FastStor 2	/dev/sg3
Tape drive	HP Ultrium 2	/dev/sg4

3. Use the `mkdev` command in `obtool` to create attach points so that Oracle Secure Backup can identify devices for backup and restore operations.

   The following example creates attach points for the tape library and tape drive shown in Table 2–5.

```
ob> mkdev -t library -o -a node1:/dev/sg3 lib1
ob> mkdev -t tape -o -a node1:/dev/sg4 -l lib1 -d 1 tape1
```

Configuring the Solaris sgen Driver to Provide Oracle Secure Backup Attach Points

Prior to Oracle Secure Backup release 10.3.0.3, Oracle Secure Backup provided a loadable kernel driver to control the library (changer) and tape (sequential) devices. Starting with Oracle Secure Backup 10.3.0.3, this kernel driver is removed. The standard sgen driver that is included with Solaris now provides the functionality provided by the kernel driver.

Enabling the Solaris sgen Driver for Changer and Sequential Devices

You must enable the Solaris sgen driver for changer and sequential devices before you install Oracle Secure Backup.

Use the following steps to enable the Solaris sgen driver for sequential and changer devices:

1. If your host does not have a previous installation of Oracle Secure Backup, skip to Step 2.

   When you enable the Solaris sgen driver on a host that already has Oracle Secure Backup installed, the attach points and device configuration are lost. You must first uninstall Oracle Secure Backup using the steps described in "Uninstalling Oracle Secure Backup on Linux or UNIX" on page 2-21.

   While uninstalling, it is recommended that you remove the `backup` directory. You can retain that `admin` directory.

2. Enable sequential (01) and changer (01) devices by adding the following line in the `/kernel/drv/sgen.conf` file:

```
device-type-config-list="sequential","changer";
```

   Note: If `device-type-config-list` is already defined for other devices, add "sequential" and "changer" to the existing list in the `sgen.conf` file.

3. Verify that there is an entry for the sgen driver in `/etc/minor_perm`.

An example of an entry in this file is as follows:

```
"sgen:* 0600 root sys"
```

4. Verify that there is an entry for the sgen driver in `/etc/name_to_major`.

   The following is an example of an entry in this file:

   ```
   "sgen 151"
   ```

5. Remove the links in `/dev/scsi/changer` and `/dev/scsi/sequential` using the following commands:

   ```
   rm -r /dev/scsi/changer
   rm -r /dev/scsi/sequential
   ```

6. Unconfigure the st driver for type 01 devices using the following command:

   ```
   update_drv -d -i '"scsiclass,01"' st
   ```

7. Configure sgen driver for the types 01 and 08 using the following command:

   ```
   add_drv -m '* 0666 bin bin' -i '"scsiclass,01" "scsiclass,08" "scsa,01.bmpt"
   "scsa,08.bmpt"' sgen
   ```

After you complete the steps to enable the sgen driver, there must be entries in `/dev/scsi/changer` for every library and `/dev/scsi/sequential` for every tape device. If you do not find these entries, restart your host system using the following commands:

```
touch /reconfigure
reboot
```

Utilizing sgen Attach Points

The entries that are made in the `/dev/scsi/changer` and `/dev/scsi/sequential` directories when you enable the Solaris sgen driver must be used as Oracle Secure Backup targets for `/dev/ob` links. These entries vary depending on the version of Solaris.

It is recommended that you create links in `/dev` in the form `/dev/obln` and `/dev/obtn` that point to the entries in `/dev/scsi/changer` or `/dev/scsi/sequential`. There must be a unique `/dev/obln` or `/dev/obtn` entry for each device that Oracle Secure Backup utilizes. These entries in `/dev` are used as attach points in the `obtool mkdev` command during Oracle Secure Backup device configuration.

Performing an Upgrade Installation on Linux or UNIX

In preparation for an upgrade, Oracle recommends that you do the following:

1. Copy your `$OSB_HOME/admin` directory to a secure but easily accessed location.
2. If you customized the `obparameters` file, then save a copy of it.
3. Cancel all active and pending jobs.
4. Stop all Oracle Secure Backup daemons.
5. Run the setup scripts from the new CD-ROM.
6. During the upgrade process, the installer displays the following prompt:

```
Oracle Secure Backup is already installed on this machine (myhostname).
Would you like to re-install it preserving current configuration data[no]?
```

Enter `yes` to perform the upgrade installation, retaining your previous configuration.

Uninstalling Oracle Secure Backup on Linux or UNIX

This section explains how to uninstall Oracle Secure Backup from a Linux or UNIX host. In this procedure Oracle Secure Backup is uninstalled from the administrative server. The procedure is the same when using the administrative server to uninstall Oracle Secure Backup from other hosts.

1. Log on as `root` to the administrative server.
2. Use the following command to identify processes related to Oracle Secure Backup:

   ```
   # /bin/ps -ef |grep ob
   ```

3. Shut down processes related to Oracle Secure Backup, such as the `http` processes for the Oracle Secure Backup Web tool.

   The appendix "Startup and Shutdown of Oracle Secure Backup Services" in *Oracle Secure Backup Reference* lists operating system-specific commands for shutting down and starting Oracle Secure Backup processes on Linux and UNIX.

   Alternatively, you can terminate `observiced`, which stops all processes. Use the following command to end each process in the list associated with Oracle Secure Backup, where *`pid`* is the process ID of `observiced`:

   ```
   kill pid
   ```

4. Change directory to the Oracle Secure Backup home directory. For example:

   ```
   # cd /usr/local/oracle/backup
   ```

   Note: If you uninstall Oracle Secure Backup from the administrative server, then the `uninstallob` script removes the Oracle Secure Backup home directory after the uninstall process.

5. Run the `uninstallob` script:

   ```
   # ./install/uninstallob
   ```

   The `uninstallob` script displays a welcome message and then asks for the name of the host from which you want to remove Oracle Secure Backup.

6. Enter the name of a host from which you want to uninstall Oracle Secure Backup.
7. The `uninstallob` script asks for the name of the `obparameters` file used for installation.

   If you created an `obparameters` file in a location other than the default, then enter the correct path information. Otherwise, press the Enter key to accept the default value `install/obparameters`.

8. The `uninstallob` script asks to remove the Oracle Secure Backup home directory. Select one of the following options:

   - `no`

Select this option if you do not want to remove the Oracle Secure Backup home directory.

- `yes`

  Select this option to remove the Oracle Secure Backup home directory. All files in the home directory are deleted. The only exception is the `admin` directory, which you can elect to retain by answering `yes` at the next prompt.

This procedure assumes you are saving the Oracle Secure Backup home directory.

9. The `uninstallob` script asks to save the Oracle Secure Backup `admin` directory, even if you have chosen not to save the entire Oracle Secure Backup home directory. Select one of these options:

   - `no`

     Select this option to remove the admin directory.

   - `yes`

     Select this option to save the admin directory. If you keep the admin directory, then you can reinstall the Oracle Secure Backup software later without destroying your administrative domain.

   This procedure assumes you are saving the Oracle Secure Backup `admin` directory.

10. The `uninstallob` script displays the choices you have made and asks to continue with the uninstallation on this host. Select one of the following options:

    - `yes`

      If you select this option, then the `uninstallob` script displays progress messages as it uninstalls Oracle Secure Backup. When it is finished, it displays the following message:

      ```
      Oracle Secure Backup has been successfully removed from host_name.
      ```

    - `no`

      If you select this option, then the `uninstallob` script does not uninstall Oracle Secure Backup from this host.

3

Installing Oracle Secure Backup on Windows

This chapter explains how to install Oracle Secure Backup on hosts that run the Windows operating system.

This chapter contains these sections:

Preliminary Steps

Perform these preliminary steps before you begin installation of Oracle Secure Backup software:

- Decide which **roles** to assign the hosts in your network, as described in "Installation and Configuration Overview" on page 1-13.
- Ensure that each host has a network connection and runs **TCP/IP**.
- If you are installing Oracle Secure Backup on a **media server**, then physically attach each **tape library** and **tape drive** that you intend to make available for use by Oracle Secure Backup. Restart the media server if required.
- Disable any system software that scans and opens arbitrary **SCSI** targets before adding a **tape device** to an **administrative domain**. If Oracle Secure Backup has to contend with other system software (such as monitoring software) for access to tape libraries and drives, then unexpected behavior can result.
- Log on to your host as either the Administrator user or as a user that is a member of the Administrators group.
- For hosts to be used in the media server role, follow the steps in "Disabling Removable Storage Service on Windows Media Servers" on page 3-2 to prevent conflicts between Oracle Secure Backup and other software on your system.

Note: If you are installing Oracle Secure Backup in an Oracle Real Application Clusters (Oracle RAC) environment, then you must install Oracle Secure Backup on each node in the cluster.

Disabling Removable Storage Service on Windows Media Servers

The Removable Storage service is used to manage removable media, drives, and libraries. On Windows hosts configured for the media server role, this service must be disabled for the Oracle Secure Backup device driver to correctly control a tape device.

To disable the Removable Storage service:

1. From the Windows Control Panel, double-click **Administrative Tools**.
2. Double-click **Services** to view the list of services on your host.
3. Right-click the **Removable Storage** service and choose **Properties**.
4. In the Properties window, if the service is running, then click **Stop** to stop the service. Set the Startup Type field to **Disabled**.
5. Click **OK**.

Extracting Oracle Secure Backup from OTN Download on Windows

If you do not have the Oracle Secure Backup distribution CD-ROM, then you must download the installation package as a Zip file from Oracle Technology Network (OTN) and extract it into a directory on your local hard drive.

To download and extract the Oracle Secure Backup installation Zip file on Windows:

1. Log on to your host as a user with Administrator privileges.
2. In Windows Explorer, create a temporary folder called osbdownload on a file system with enough free space to hold the downloaded installation file.
3. Open a Web browser and go to the Oracle Secure Backup Web site on Oracle Technology Network (OTN):

   `http://www.oracle.com/technetwork/products/secure-backup/overview/index.html`
4. Click the Downloads tab.

   The displayed page contains the following two sections: Oracle Secure Backup Downloads and Oracle Secure Backup Express Downloads.
5. Click the version of Oracle Secure Backup that you want to download.

   The Oracle Secure Backup Downloads page is displayed.
6. Click **OTN License Agreement**.

   The Oracle Technology Network Developer License Terms page is displayed in a separate browser window.
7. Read the **Export Controls on the Programs** and close the window.

   You must accept the OTN License Agreement to download the Oracle Secure Backup software.
8. Select the **Accept License Agreement** option, and click the link for the version of Oracle Secure Backup release 10.4 specific to your operating system.

> **Note:** If you have multiple operating systems in your environment, then you must perform multiple downloads of the Oracle Secure Backup release 10.4 software.

9. The Oracle Secure Backup release 10.4 installation software is compressed. Save it to a temporary directory, and expand it to the osbdownload directory you created in step 2.

You now have all of the files required to install Oracle Secure Backup release 10.4.

Running the Oracle Secure Backup Windows Installer

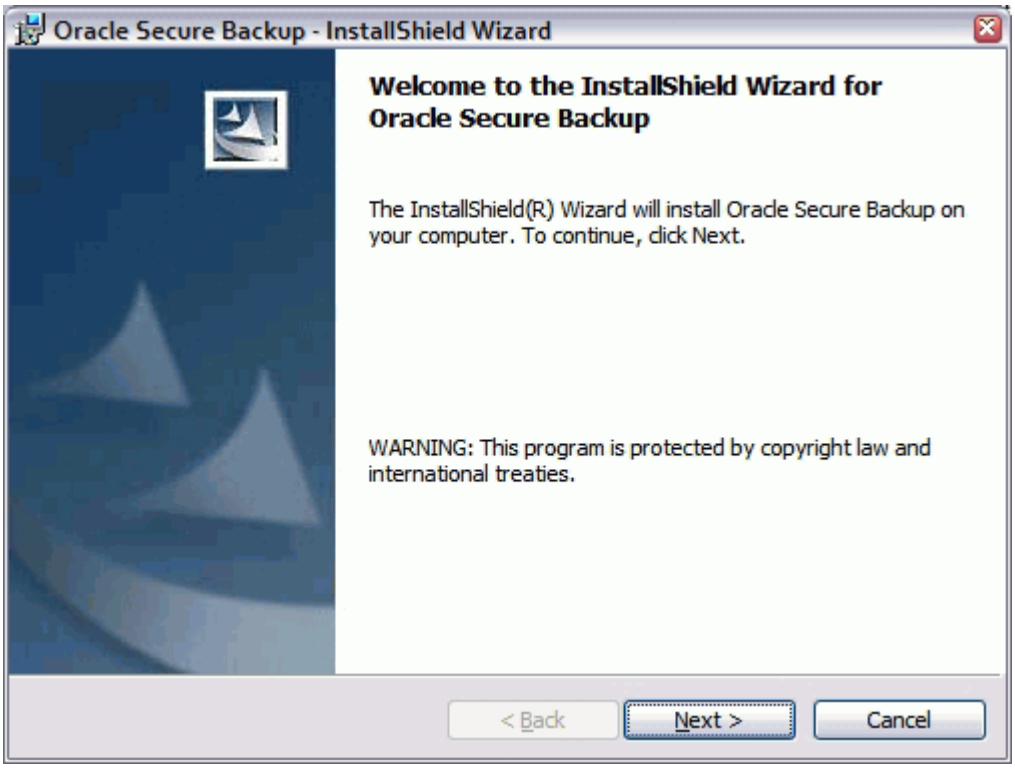
Complete the following steps to install Oracle Secure Backup on a Windows host:

> **Note:** If you are installing Oracle Secure Backup in an Oracle RAC environment, then you must install Oracle Secure Backup on each node in the cluster.

1. Select one of these install options:
   - If you are installing Oracle Secure Backup from a CD-ROM, then insert the CD-ROM. If AutoPlay is enabled, then the setup.exe program starts automatically and opens the Oracle Secure Backup Setup Wizard.

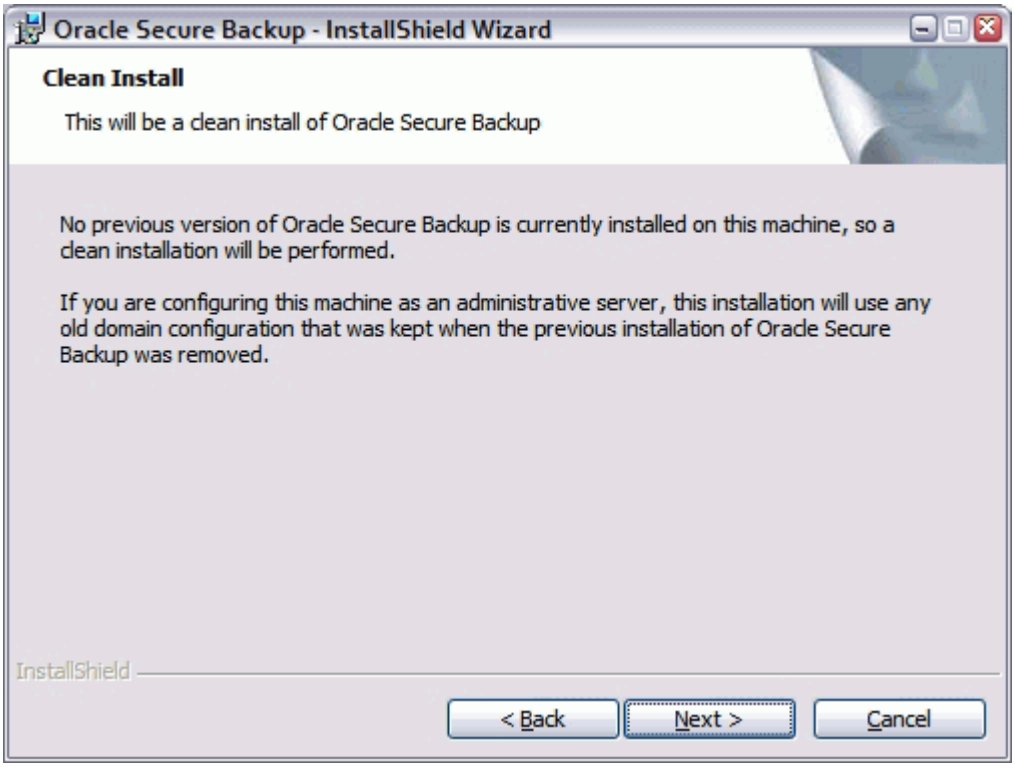
     If Windows AutoPlay is not enabled, then open the drive containing the installation CD-ROM using Windows Explorer and run the setup.exe program.
   - If you are installing Oracle Secure Backup from an Oracle Technology Network (OTN) download, then run the setup.exe program from the folder into which the download Zip file contents were extracted.

   The Oracle Secure Backup Setup Wizard starts and the Welcome screen appears.



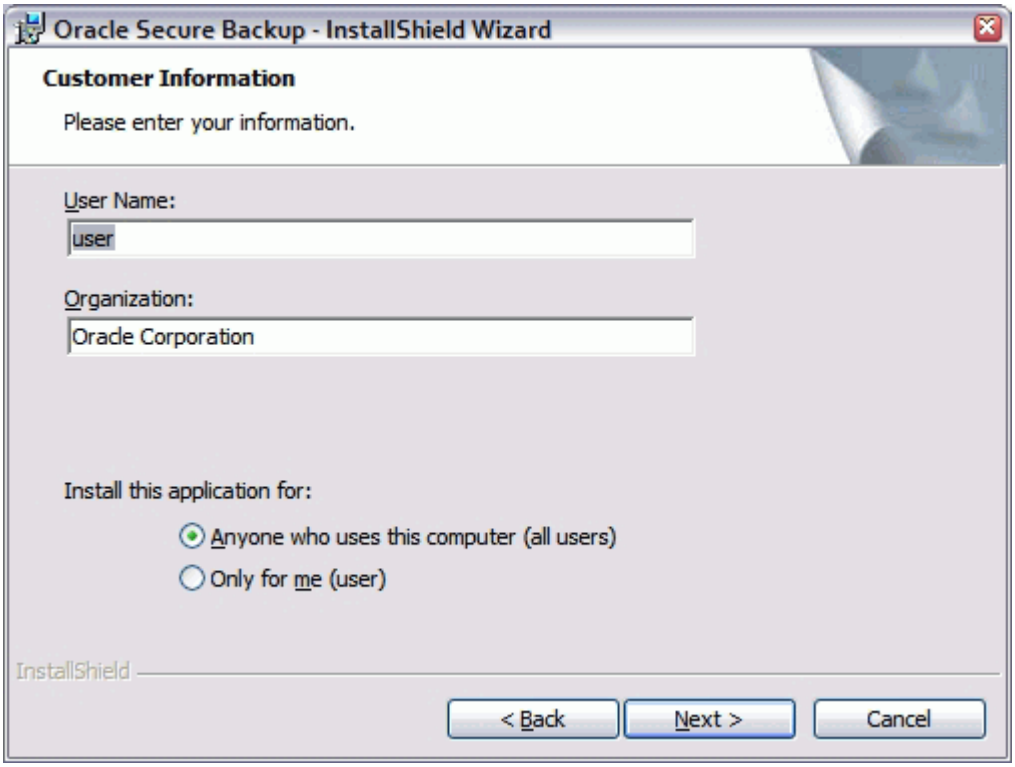
2. Click **Next** to continue.

   If you have uninstalled Oracle Secure Backup software before beginning this installation, or if you have never installed it on this computer, then the Clean Install page appears.



3. Click **Next** to continue.

   The Customer Information screen appears.



4. Enter your customer information as follows:

   a. Enter a user name in the **User Name** field.

   b. Enter the name of your company in the **Organization** field.

   c. Select one of these options:

      – **Anyone who uses this computer**

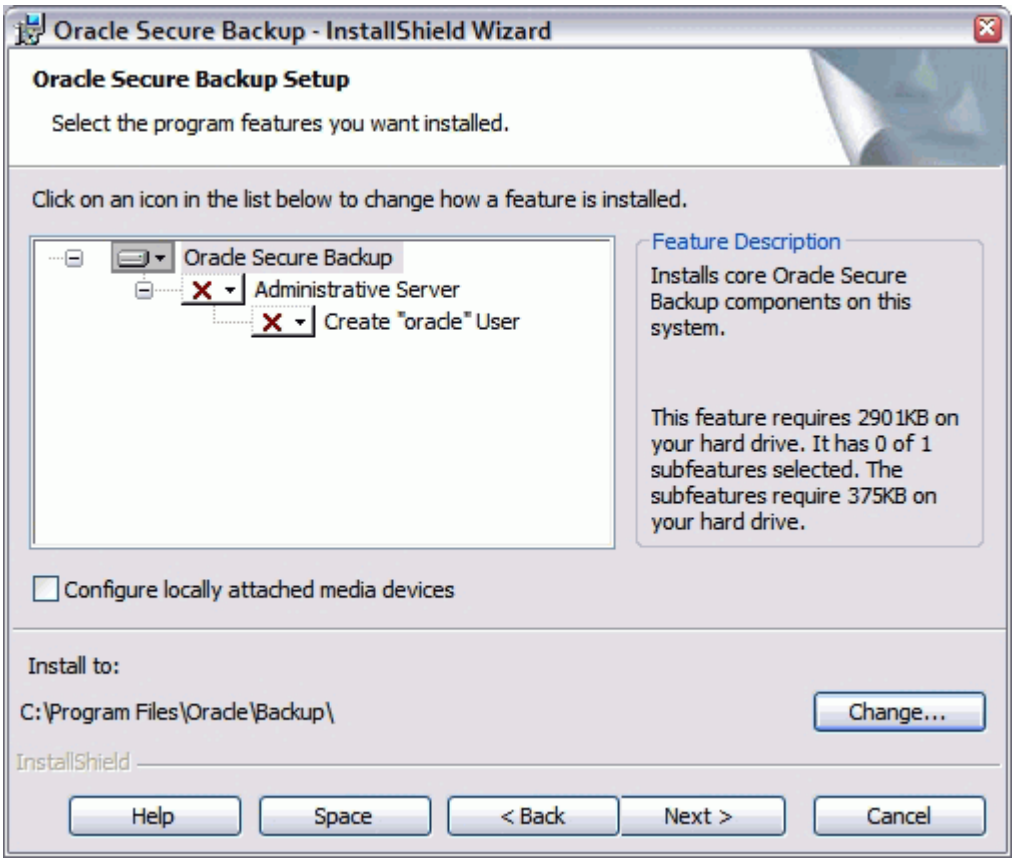
        This option allows anyone who has access to this computer to use Oracle Secure Backup.

      – **Only for me**

        This option limits use of Oracle Secure Backup to you.

   Click **Next** to continue.

   The Oracle Secure Backup Setup screen appears.



5. A single host can have multiple roles, which are are additive rather than exclusive. You have the following options when choosing roles:

   - To install the Windows host as client only, click **Next** and go to step 9.

     Note: Every installation of Oracle Secure Backup on Windows includes a client installation.

   - If you want this Windows host to serve as a media server, then select the **Configure locally attached media devices** option, click **Next**, and go to step 9.

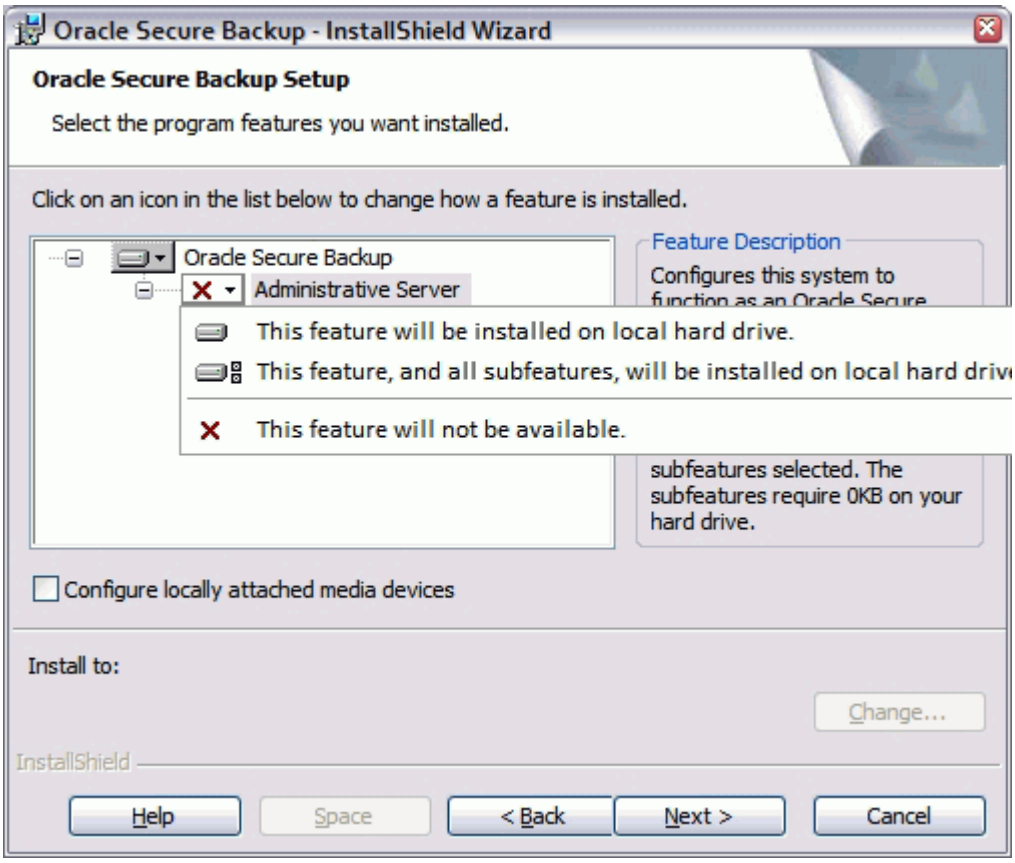
     Oracle Secure Backup always installs the *software* required for the media server role. But if you want this Windows host to have the media server *role* in your Oracle Secure Backup administrative domain, then you must complete the Oracle Secure Backup software installation, configure any tape devices attached to this host, and then add the media server role.

     If you select the **Configure locally attached media devices** option, then the Oracle Secure Backup Configuration utility enables you to configure the tape devices attached to this computer. If you do not select this option, then the Oracle Secure Backup Configuration utility ignores any attached tape devices.

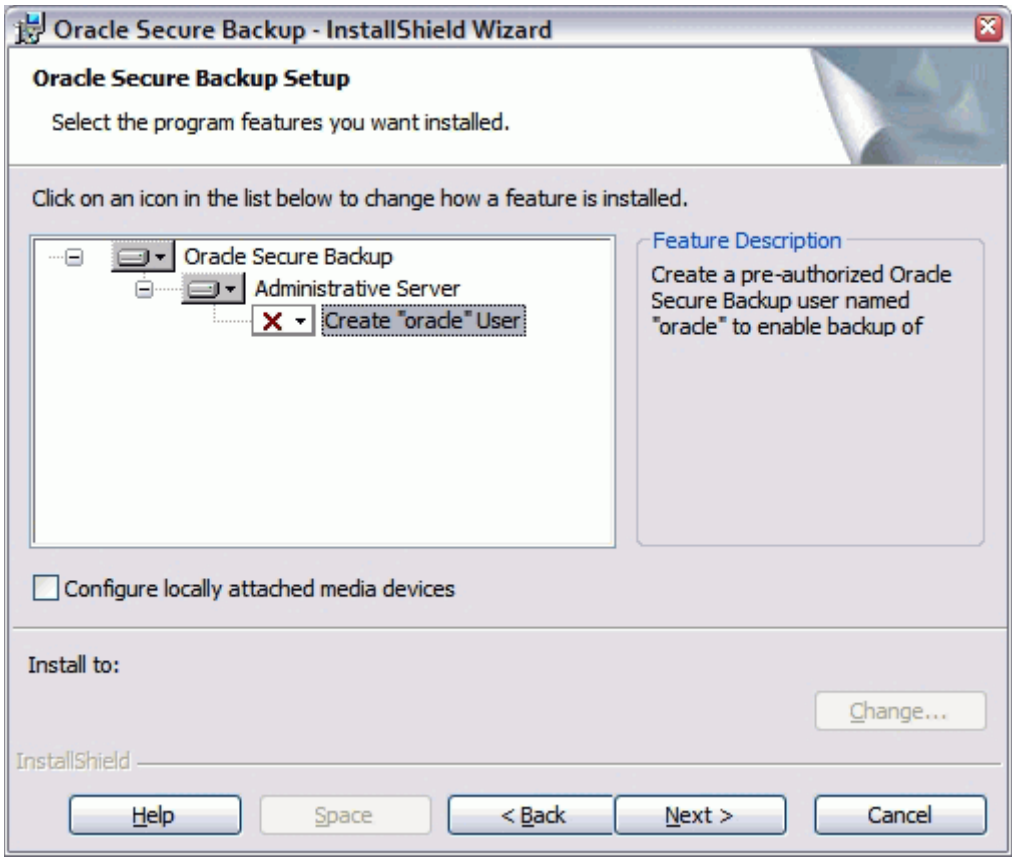
See Also:

- "Configuring Oracle Secure Backup" on page 3-14
- Chapter 5, "Configuring and Managing the Administrative Domain"

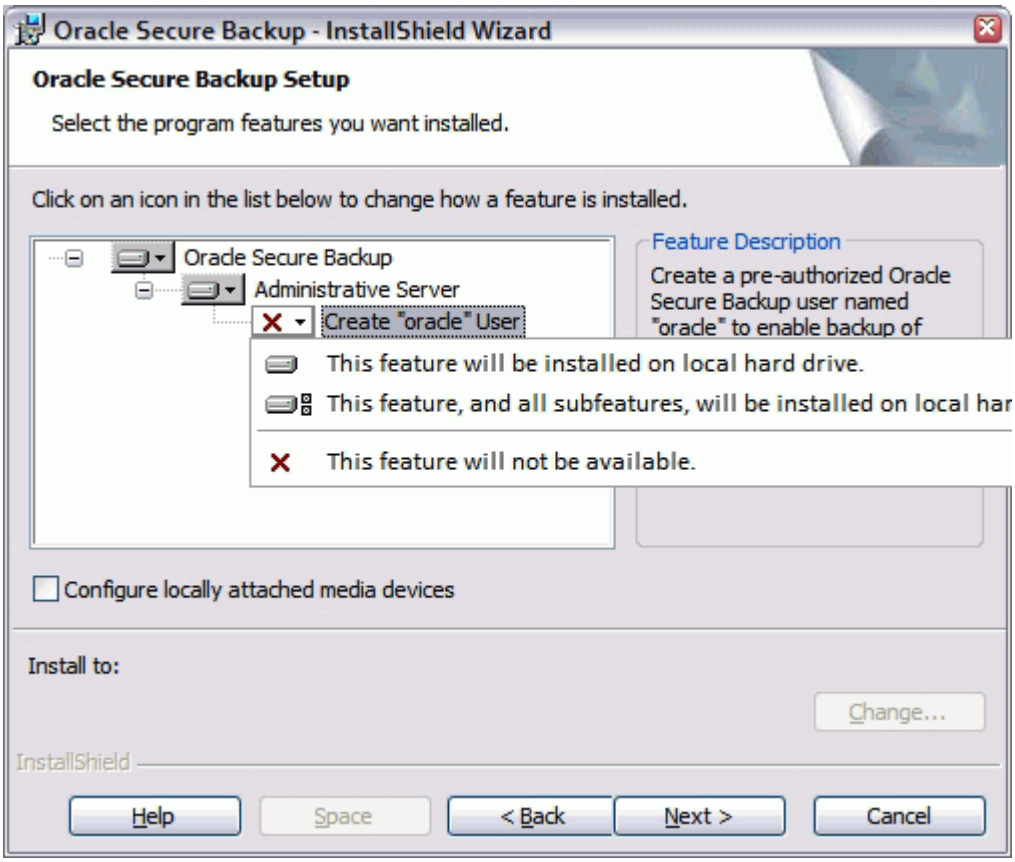
- To install the Windows host as an **administrative server**, click the Administrative Server list and select **This feature will be installed on local hard drive**.



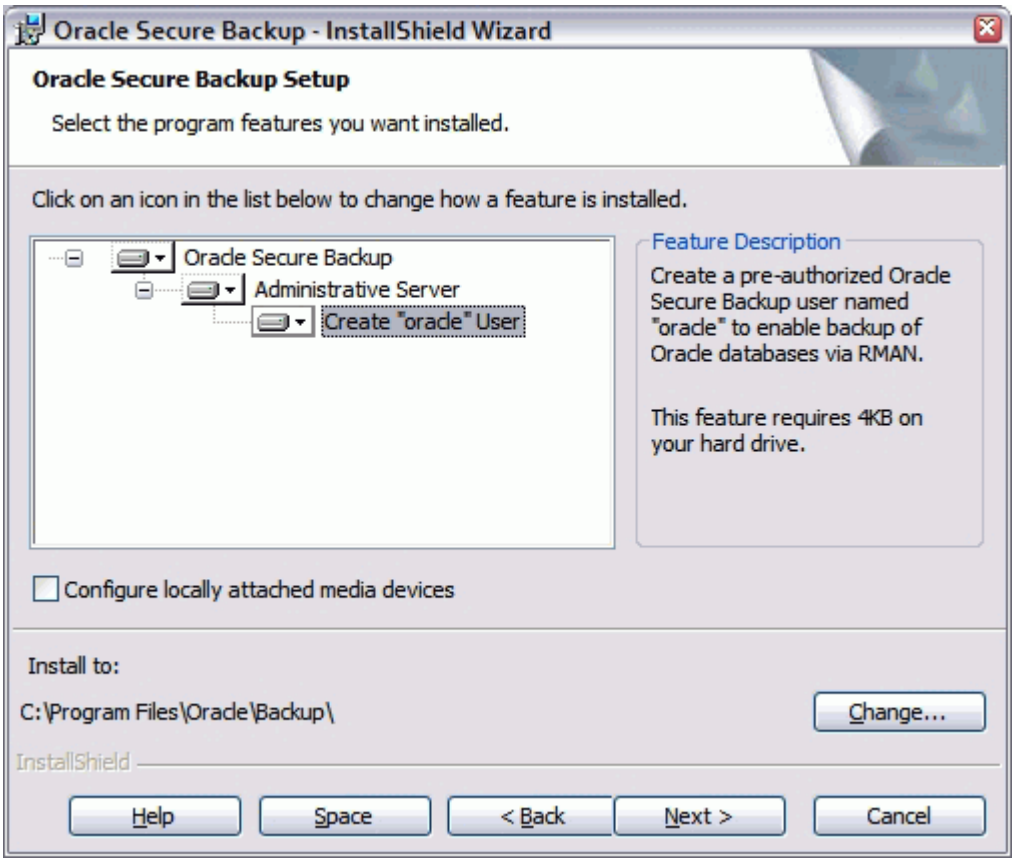
Selecting this option removes the X from the administrative server icon and includes the administrative server role in the installation.



6. If you plan to perform Oracle Database backup and restore operations with RMAN, then enable the action for **Create "oracle" user** in the administrative server submenu.



If this option is enabled, then the installer creates an **Oracle Secure Backup user** called `oracle` (with the **rights** of the `oracle` **class**) whose purpose is to facilitate Oracle Database backup and restore operations with **Recovery Manager (RMAN)**.



Note:

- You are required to create the `oracle` user only if you plan to use Oracle Secure Backup with RMAN.
- If you intend to use Oracle Secure Backup to perform one-time, RMAN-initiated, or **unprivileged backup** operations on Windows clients, then you must modify the Oracle Secure Backup `admin` and `oracle` users to assign them Windows credentials (a domain, user name and password) that are valid at the client with required privileges after you complete the Oracle Secure Backup installation. Otherwise, Oracle Secure Backup cannot perform the backup operation. This requirement applies regardless of the platform that acts as the administrative server.
- The installer assigns a random password to the `oracle` user. In most cases you are not required to change the assigned password, because it is not usually necessary to log in to Oracle Secure Backup using this user account.
- Before electing to create an Oracle Secure Backup `oracle` user, be aware that this choice involves a trade-off between convenience and security.

See Also: *Oracle Secure Backup Reference* for more information about the `oracle` **class**

If you do not plan to use Oracle Secure Backup to back up your databases, then leave the **Create "oracle" user** option unselected. This is the default.

In addition to the options described in step 6, you can perform the following actions in the Oracle Secure Backup Setup screen:

- Click **Help** for detailed descriptions of the installation options.
- Click **Change** to change the destination folder for the installation.
- Click **Space** to display the disk space required for the installation.

Click **Next** to continue.

The Oracle Secure Backup Encryption Key Store Password screen appears.

7. Enter a password for the Oracle Secure Backup encryption wallet in the **Password for encryption wallet** field.

   Enter the password again in the **Re-type password for verification** field.

   Click **Next**.

   The Oracle Secure Backup Admin User Password and Email screen appears.

8. Enter a password for the Oracle Secure Backup `admin` user in the **Password for 'admin' user** field.

   Enter the password again in the **Re-type password for verification** field.

   The minimum password length is determined by the `minuserpasswordlen` security policy. Its value at installation time is `0`, which means a null password is permitted. After the installation has completed, you can change this policy to enforce a different minimum password length.

   > **See Also:** *Oracle Secure Backup Reference* for more information on the `minuserpasswordlen` security policy

   ---

   Note: Oracle suggests that you choose an administrative user password of at least eight characters in length, containing a mixture of alphabetic and numeric characters. The maximum length is 16 characters.

   The practice of supplying a password in clear text on a command line or in a command script is not recommended by Oracle. It is a security vulnerability. The recommended procedure is to have the user be prompted for the password.

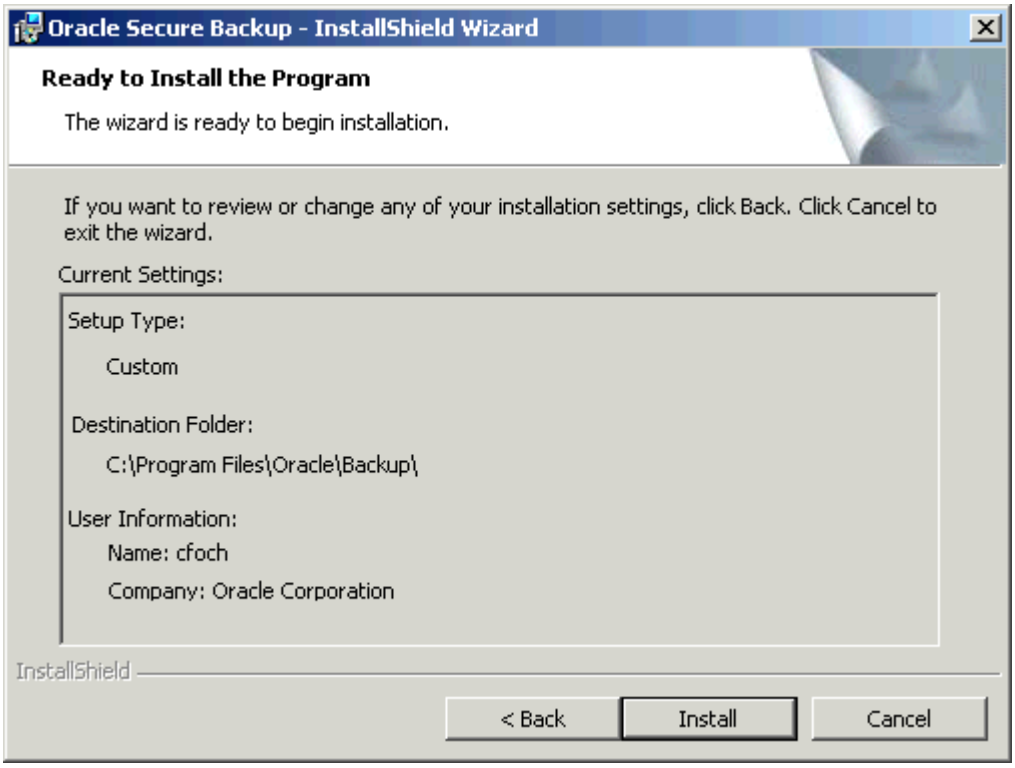
   Enter an e-mail address in the **Email address for 'admin' user:** field.

   Entering an email address for the `admin` user enables Oracle Secure Backup to send notifications of important events. Setting this field is optional.

Note: The default *from* address for e-mails generated by Oracle Secure Backup is `SYSTEM@`*`fqdn`*, where fqdn is the fully qualified domain name of the Oracle Secure Backup administrative server. You can change this default *from* address after installation. See *Oracle Secure Backup Reference* for more information.

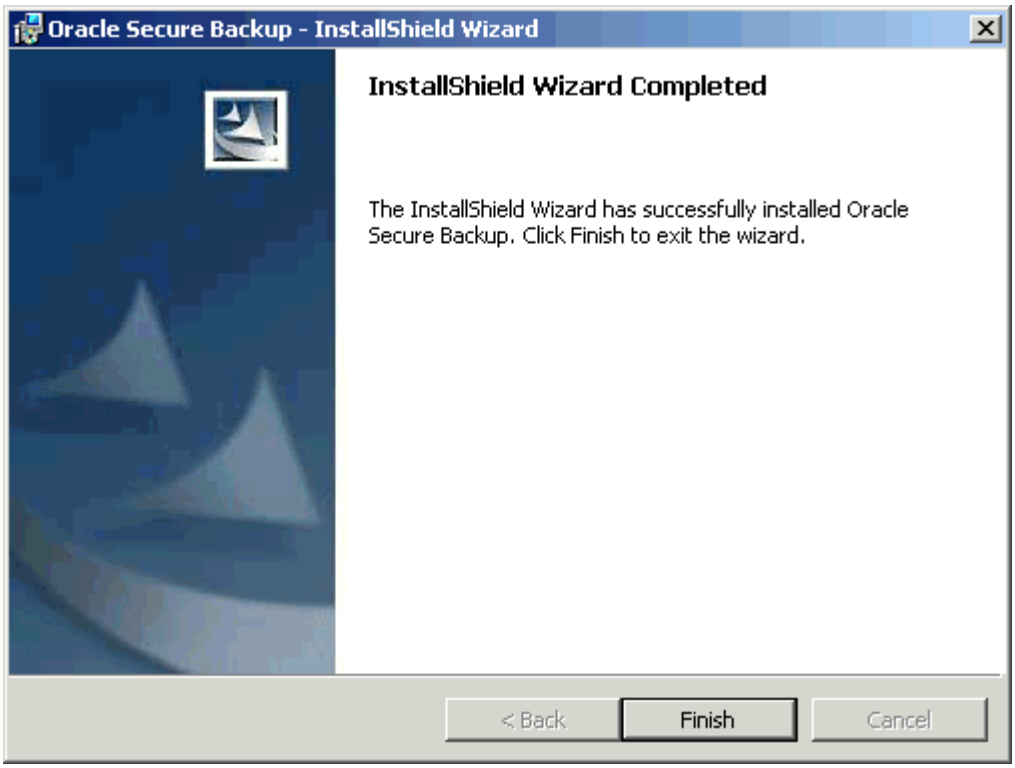
Click **Next**.

The Ready to Install the Program screen appears.



9. Click **Install** to start copying files.

   A progress bar appears. When the files are copied the InstallShield Completed screen appears.



10. Click **Finish**.

    The Oracle Secure Backup software installation on this Windows host is complete. You can now configure this installation, using the Oracle Secure Backup Configuration utility that starts automatically. Instructions on using this utility appear in "Configuring Oracle Secure Backup" on page 3-14.

Configuring Oracle Secure Backup

This section explains how to configure Oracle Secure Backup using the Oracle Secure Backup Configuration utility. This utility starts automatically when you click Finish on the final Installshield Wizard screen during the installation of Oracle Secure Backup.

If you complete this initial configuration and subsequently want to view or change your configuration settings, then you can revisit the Oracle Secure Backup Configuration utility in either of two ways:

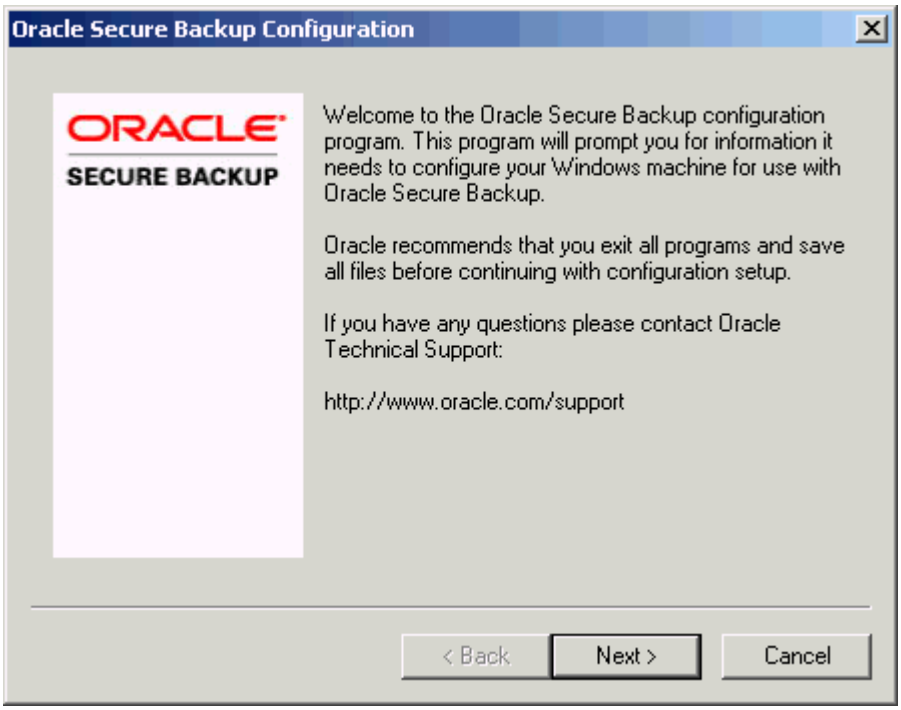
- Select **Start** > **All Programs** > **Oracle Secure Backup** > **Oracle Secure Backup Configuration**
- Enter `obcfg` at the command line

Complete the following steps to configure Oracle Secure Backup on a Windows host:

1. Start the Oracle Secure Backup Configuration utility.

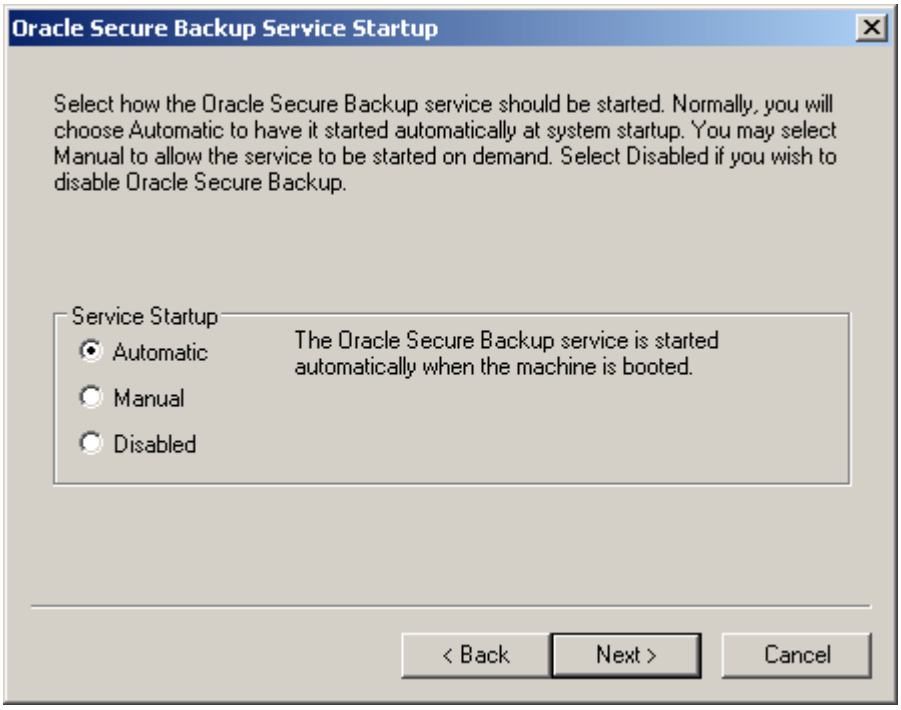
   Note: This step is unnecessary if you are configuring Oracle Secure Backup on a Windows host for the first time, because the Oracle Secure Backup Configuration utility starts automatically after the Oracle Secure Backup software installation process.

   The Oracle Secure Backup Configuration welcome screen appears.



2. Click **Next**.

   The Oracle Secure Backup Service Startup screen appears.



3. Select one of these modes in which to start the Oracle Secure Backup service:

   - **Automatic**

     The Oracle Secure Backup service starts automatically when you restart your host.

- **Manual**

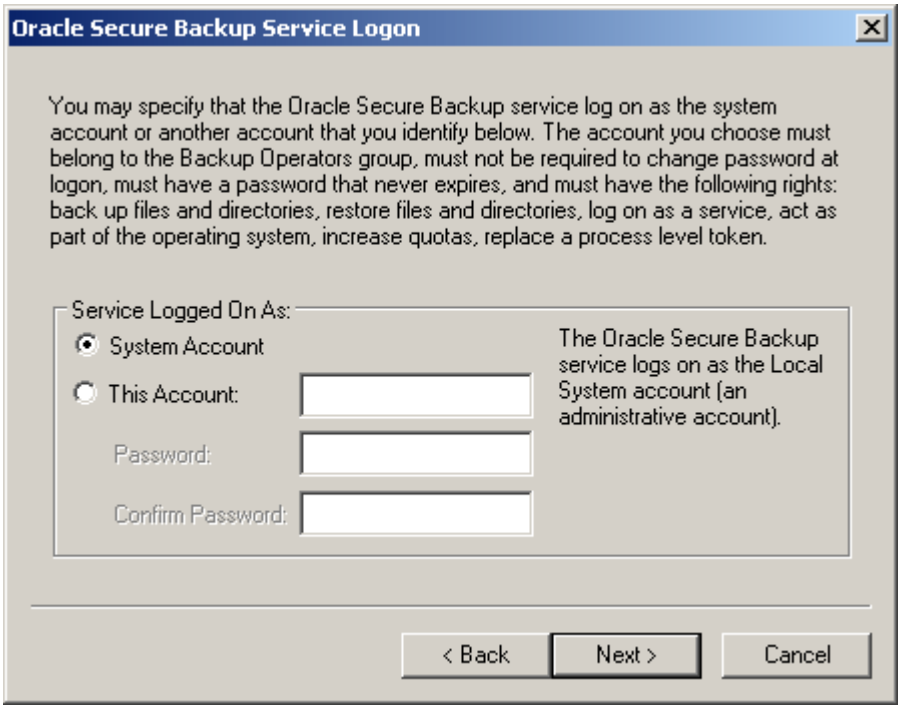
  The Oracle Secure Backup service must be started manually by a user who is a member of the Administrators group.

- **Disabled**

  The Oracle Secure Backup service is disabled.

Click **Next**.

The Oracle Secure Backup Service Logon screen appears.



4. By default, the Oracle Secure Backup service logs on as the Local System account, which is an administrative account. You can select option **This Account** to specify a different account for the Oracle Secure Backup Service.

   Select one of these options:

   - **System Account**

     Select this option if you plan to run the Oracle Secure Backup **service daemon** (and associated subordinate **daemons**) with full privileges.

   - **This Account**

     Select this option if you plan to run the Oracle Secure Backup service daemon (and associated subordinate daemons) with the privilege set associated with an existing Windows user account. You must fill in the Windows user account name and password.

     If you choose this option, then you must ensure that the Windows user account you select meets the following criteria:

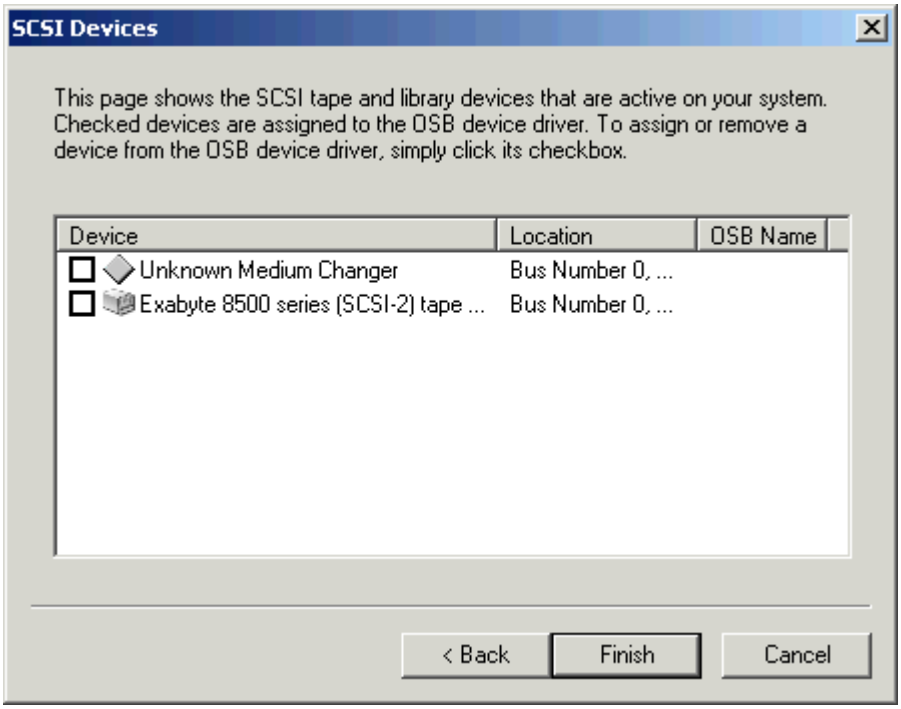
     – The account you choose must belong to the Backup Operators group.

     – No change in password at login is required of the account.

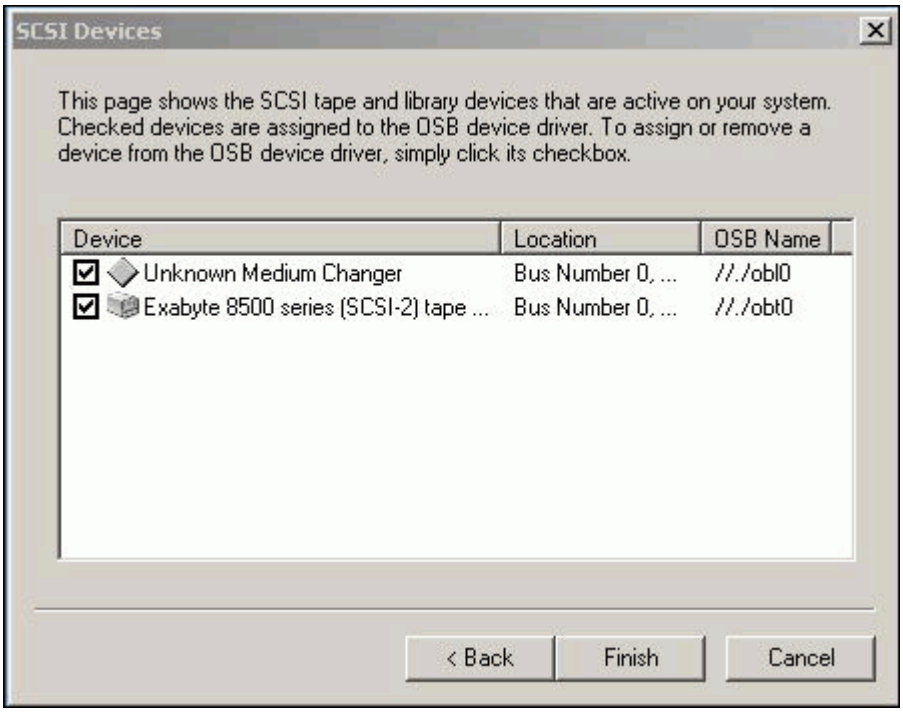
     – The account must be set so that the password never expires.

- The account must have backup and restore rights.
- The account must be able to restore files and directories.
- The account must be able to log on as a service.
- The account must be able to act as part of the operating system.
- The account must be able to increase quotas.
- The account must be able to replace a process level token.

Click **Next** or **Finish** to proceed. If you are configuring a media server, then proceed to step 5.



5. Select the tape library and tape drive to assign to the Oracle Secure Backup device drivers. After a short delay, the devices are redisplayed with check marks in the first column and an Oracle Secure Backup device name for each of them in the last column. Make a note of the device name assigned to each device. You must have these device names when you set up the devices in Oracle Secure Backup later on.



6. Click **Finish**.

   When you have performed all of the preceding tasks, Oracle Secure Backup installation and configuration on this host is complete. Repeat this installation and configuration process for each Windows host in your administrative domain.

Configuring Firewalls for Oracle Secure Backup on Windows

Windows XP Service Pack 2 and Windows Server 2003 contain a built-in Windows Firewall which, in the default configuration, blocks inbound traffic on ports used by Oracle Secure Backup.

If your Windows host is protected by a **firewall**, then the firewall must be configured to permit Oracle Secure Backup **daemons** on the host to communicate with the other hosts in your administrative domain. Oracle Secure Backup includes daemon components that listen on port 400, port 10000, and other dynamically assigned ports.

Because the dynamically assigned ports used by Oracle Secure Backup span a broad range of port numbers, your firewall must be configured to allow executables for the Oracle Secure Backup daemons to listen on all ports.

The Oracle Secure Backup Windows installation provides a sample batch script called obfirewallconfig.bat in the bin directory under the Oracle Secure Backup home.

This script contains commands that make the required configuration changes for the Windows Firewall on Windows Server 2003 and Windows XP systems having a single network interface. Review the script to determine whether it is suitable for your environment. You can run the script after the installation completes.

For details on configuration of other firewalls, see the documentation provided by the vendor. You can refer to the sample script for the Windows Firewall to determine the names of executables that need permission to listen on ports.

Performing an Upgrade Installation on Windows

Use the following steps to upgrade your Windows 32-bit or Windows 64-bit hosts to Oracle Secure Backup 10.4.0.3:

1. Uninstall the existing Oracle Secure Backup software as described in "Uninstalling Oracle Secure Backup on Windows" on page 3-19.

   While uninstalling your existing Oracle Secure Backup software, you must select the **Keep** option to retain the existing configuration of your administrative domain.

   Note: While upgrading a media server or an administrative server that is also a media server from Oracle Secure Backup 10.1 or Oracle Secure Backup 10.2, ensure that you restart the host after you uninstall the existing software.

2. Run the Oracle Secure Backup release 10.4 installer to install the software as described in "Installing Oracle Secure Backup on Windows" on page 3-1.

Uninstalling Oracle Secure Backup on Windows

Complete the following steps to uninstall Oracle Secure Backup on Windows:

1. Select **Start** > **All Programs** > **Oracle Secure Backup** > **Uninstall Oracle Secure Backup**.

   A confirmation dialog appears.

2. Click **Yes** to remove Oracle Secure Backup from your computer.

3. If you configured your host as an administrative server, then an additional window opens asking whether you want to preserve the files specific to your administrative domain. Select one of these options:

   - Click **Delete** if you do not want to retain the administrative domain files.
   - Click **Keep** to retain the administrative domain files.

     If you click **Keep** to retain the administrative domain files, then the configuration of your administrative domain is preserved. This is useful for reinstallation of the Oracle Secure Backup software later.

Oracle Secure Backup is now uninstalled from your host.

4

Oracle Secure Backup User Interfaces

This chapter introduces the interfaces that you can use with Oracle Secure Backup. The major interfaces to Oracle Secure Backup are:

- Oracle Enterprise Manager

  This is the primary graphical user interface for managing Oracle Secure Backup.

- Oracle Secure Backup **Web tool**

  This interface is used to manage file-system level backups and to perform certain other tasks not possible in Oracle Enterprise Manager.

- **obtool**

  This command line client exposes the full functionality of Oracle Secure Backup and is invoked by the Oracle Secure Backup Web Tool and Oracle Enterprise Manager.

Note:

- Database backups are performed using **Recovery Manager (RMAN)**. Because backup and recovery activities are discussed in *Oracle Secure Backup Administrator's Guide* and *Oracle Database Backup and Recovery Advanced User's Guide*, RMAN is not discussed in this chapter.
- All backup and restore operations in Oracle Secure Backup ultimately call upon a command line tool called **obtar**. It is generally not necessary to call obtar directly. See *Oracle Secure Backup Reference* for more details about obtar.

This chapter contains these sections:

Using Oracle Secure Backup in Enterprise Manager

You can use Oracle Enterprise Manager 10*g* (10.2) or Oracle Enterprise Manager 11*g* to perform most Oracle Secure Backup tasks, including **administrative domain** and hardware configuration, managing your media, and backing up and restoring databases. Oracle Enterprise Manager is the preferred Web interface for Oracle Secure Backup tasks.

However, you cannot use Oracle Enterprise Manager to perform **file-system backup** and restore operations. The Maintenance page in Oracle Enterprise Manager includes a link to the Oracle Secure Backup **Web tool** for such tasks.

This document describes the use of Oracle Enterprise Manager for most tasks, and describes the Oracle Secure Backup Web Tool only when there is no equivalent functionality in Enterprise Manager.

This section contains these topics:

- Enabling Oracle Secure Backup Links in Oracle Enterprise Manager
- Registering an Administrative Server in Oracle Enterprise Manager
- Accessing the Web Tool from Enterprise Manager

Enabling Oracle Secure Backup Links in Oracle Enterprise Manager

If you are using releases 10.2.0.1 or 10.2.0.2 of Oracle Enterprise Manager Grid Control or release 10.2.0.2 of Oracle Enterprise Manager Database Control, then the Maintenance page does not include the Oracle Secure Backup section by default. If the Oracle Secure Backup section does not appear in the Maintenance page, then you must configure Oracle Enterprise Manager to enable the links.

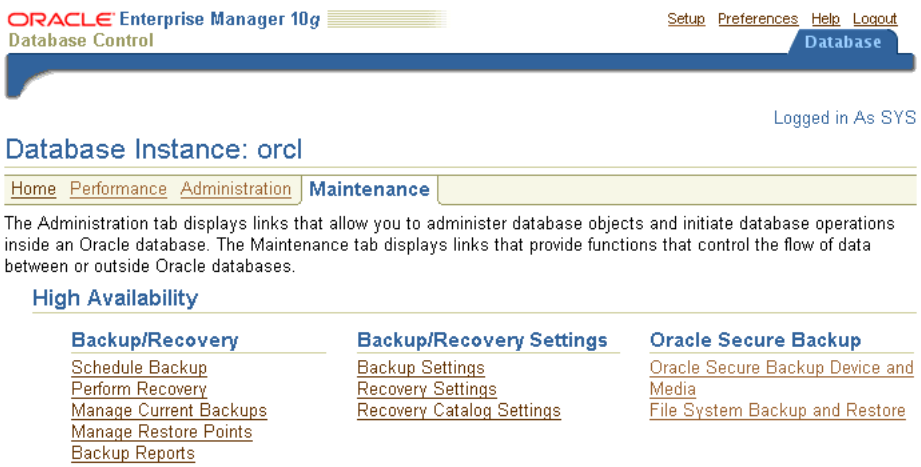
To enable the Oracle Secure Backup section in Oracle Enterprise Manager:

1. Go to the *ORACLE_HOME*`/`*hostname_SID*`/sysman/config` directory and open the emoms.properties file in a text editor.
2. Set `osb_enabled=true` and save the file.
3. Stop and restart the Oracle Enterprise Manager Database Control console with the `emctl` command:

   ```
   emctl stop dbconsole
   emctl start dbconsole
   ```

4. Go to the Maintenance page and confirm that the Oracle Secure Backup section appears, as shown in Figure 4–1.

Figure 4–1 Maintenance Page



Registering an Administrative Server in Oracle Enterprise Manager

You can make RMAN backups to the Oracle Secure Backup **SBT interface** three ways:

- Oracle Enterprise Manager Database Control
- Oracle Enterprise Manager Grid Control
- RMAN command-line client

The Database Control console must run on the **administrative server** and can only back up an Oracle database on the administrative server. You can run the Grid Control console on any database host in the administrative domain and use it to back up any database. This section describes how to get started with Database Control.

To use Enterprise Manager to manage your backups, you must make Enterprise Manager aware of your administrative server, which stores the configuration data and **catalog** for the Oracle Secure Backup administrative domain. To register the administrative server in Oracle Enterprise Manager Database Control:

1. Log in to the Oracle Enterprise Manager Database Control console as a user with database administrator **rights**.
2. In the Oracle Secure Backup section, click **Oracle Secure Backup Device and Media**.

   The Add Administrative Server page appears.
3. Log in to your Oracle Secure Backup administrative domain as follows:
   a. Enter the **Oracle Secure Backup home** directory in the **Oracle Secure Backup Home** field. This directory is usually `/usr/local/oracle/backup` on UNIX and Linux and `C:\Program Files\Oracle\Backup` on Windows.
   b. Enter the name of an Oracle Secure Backup administrative user in the **Username** field. For example, enter `admin`.
   c. Enter the password for the Oracle Secure Backup administrator in the **Password** field.
   d. Click **OK**.

   The Host Credentials page appears.
4. Enter the username and password of the operating system user on the administrative server. This user needs `root` privileges.

   The Oracle Secure Backup Device and Media: Administrative Server: *hostname* page appears. You can use this page to load tapes.

After you have registered the administrative server, you are ready to use Oracle Enterprise Manager with Oracle Secure Backup.

> **See Also:** *Oracle Database 2 Day DBA* for an introduction to using Oracle Enterprise Manager for database backup and recovery with RMAN

Accessing the Web Tool from Enterprise Manager

The Oracle Enterprise Manager console for a database provides a link to the Oracle Secure Backup Web tool. You can use this link when you need access to Oracle Secure Backup Web tool functions, such as file-system backup information.

To access the Oracle Secure Backup Web tool through Oracle Enterprise Manager Database Control:

1. Log in to the Oracle Enterprise Manager Database Control as a user with database administrator **rights**.
2. Go to the Oracle Secure Backup section of the Maintenance page.

   If the Oracle Secure Backup section does not appear in the Maintenance page, then see "Enabling Oracle Secure Backup Links in Oracle Enterprise Manager" on page 4-2.
3. Click **File System Backup and Restore**.

   The Oracle Secure Backup Web tool interface opens, as described in "Starting a Web Tool Session" on page 4-4.

Using the Oracle Secure Backup Web Tool

The Oracle Secure Backup Web tool is a browser-based interface that does not require installation of Oracle Enterprise Manager. It is also the only graphical interface to the file-system backup capabilities of Oracle Secure Backup.

> **Note:** You can access all functionality of Oracle Secure Backup through the Oracle Secure Backup Web Tool, including file-system level backups. However, Oracle Enterprise Manager is the preferred interface for most functionality, and provides the only graphical interface for Oracle Database backups to tape.

You can access the Oracle Secure Backup Web tool from any supported browser that can connect to the **administrative server** through **SSL**. The **Apache Web server** supplied with Oracle Secure Backup must be running to respond to these requests. Supported browsers are listed on Certify on My Oracle Support, at the following URL:

`http://support.oracle.com/`

> **Note:** The PHP software installed with Oracle Secure Backup is not supported for direct use by customers. It is only supported for use in implementing the Oracle Secure Backup Web tool.

This section contains these topics:

- Starting a Web Tool Session
- Web Tool Home Page
- Web Tool Configure Page
- Web Tool Manage Page
- Web Tool Backup Page
- Web Tool Restore Page

Starting a Web Tool Session

This section explains how to use the Oracle Secure Backup **Web tool** to access your Oracle Secure Backup **administrative domain**.

To start an Oracle Secure Backup Web tool session:

1. Launch your Web browser and supply the URL of the host running Oracle Secure Backup. Use the following syntax, where *hostname* can be a fully qualified domain name:

   ```
   https://hostname
   ```

   For example, you might invoke the following URL:

   ```
   https://osblin1.oracle.com
   ```

2. The browser displays a warning that the **certificate** is not trusted. Oracle Secure Backup installs a self-signed certificate for the **Apache Web server**. The Web server requires a signed certificate for data encryption purposes. The security warning appears because the browser does not recognize the signer as a registered **Certification Authority (CA)**. This alert does not mean that your data is not encrypted, only that the CA is not recognized.

   Accept the certificate. It is not necessary to view the certificate or make any configuration changes.

   The Oracle Secure Backup Login page appears.

3. Enter an **Oracle Secure Backup user** name in the **User Name** box and a password in the **Password** box.

   If you are logging into the Oracle Secure Backup Web tool for the first time, then log in as the `admin` user. You can create additional users after you log in.

   Note: Oracle recommends that you not use browser-based password managers to store Oracle Secure Backup passwords.

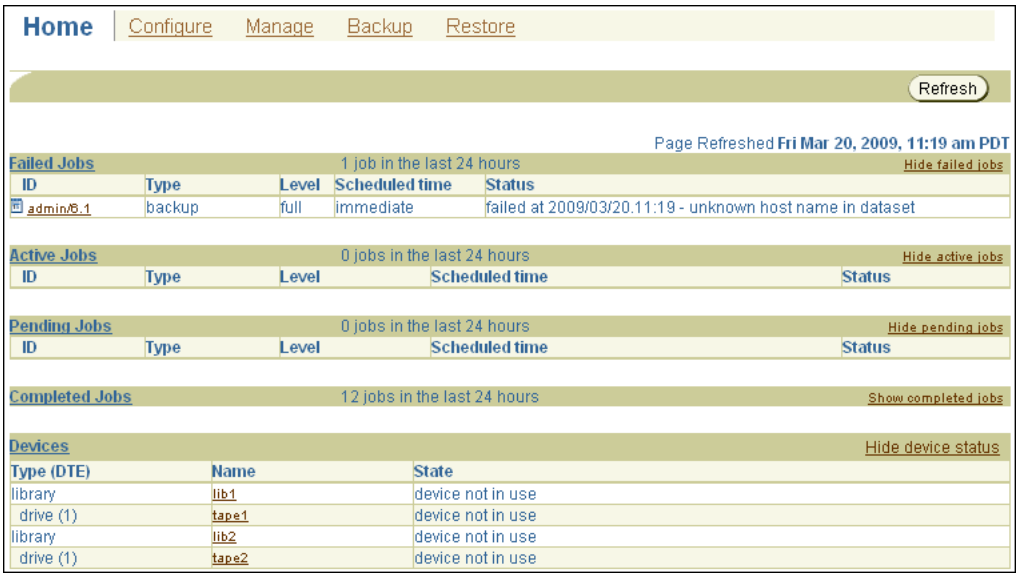
4. Click **Login**. The Oracle Secure Backup Home page appears.

   The **Home**, **Configure**, **Manage**, **Backup**, and **Restore** tabs are explained in detail in the following sections.

Web Tool Home Page

After you log in to the Oracle Secure Backup **Web tool** interface, the Oracle Secure Backup Home page appears. This page provides a summary of the current status of each Oracle Secure Backup job and **tape device**. Figure 4–2 shows an example of the Home page.

Figure 4–2 Oracle Secure Backup Home Page



The main page includes the schedule times, status, job IDs, job type, and job level of recent jobs. Oracle Secure Backup provides a link for failed jobs, alerting users and administrators to potential trouble spots.

The **Devices** link lists the tape devices associated with each job along with information concerning tape device type, device name, and status. This page provides you with an overall picture of the various backup or restore processes that are going on.

Note: A status of "device not in use" means that the tape device is present but is not currently being utilized for backup or restore operations.

A menu bar at the top of the Oracle Secure Backup Home page enables you to select among the **Configure**, **Manage**, **Backup**, and **Restore** tabs.

Note: When using the Oracle Secure Backup Web tool, ensure that your browser is configured to reload the page every time it is viewed. Otherwise, the browser might display stale information. For example, changes made in **obtool** might not be visible in the browser.

Persistent Page Links

The top and bottom panels of the Home page, and every page of the Oracle Secure Backup **Web tool** interface, have the following persistent links:

- **Help**

  Use this link to access online documentation for Oracle Secure Backup in PDF format.

- **Logout**

Logs the current user out of the Oracle Secure Backup Web tool, clears user name and password cookies, and returns to the Login page.

- **Preferences**

  Use this link to access settings for the following options:

  – Extended command output

    This option displays **obtool** commands used to perform actions and generate output pages for the Oracle Secure Backup Web Tool at the bottom of each page.

  – Background timeout

    This option sets the maximum idle time for `obtool` background processes used by the Oracle Secure Backup Web tool to retain state information across requests.

    Operations such as **catalog** browsing, data restore operations, and **on-demand backup** operations use a background `obtool` process to retain state information across HTTP requests. When the time between requests exceeds this limit, the process exits gracefully and the associated user's session state is lost. The default is 24 hours.

  – Select table size

    This option sets the number of rows in the display window of the Oracle Secure Backup Web tool interface. The default is 8 rows.

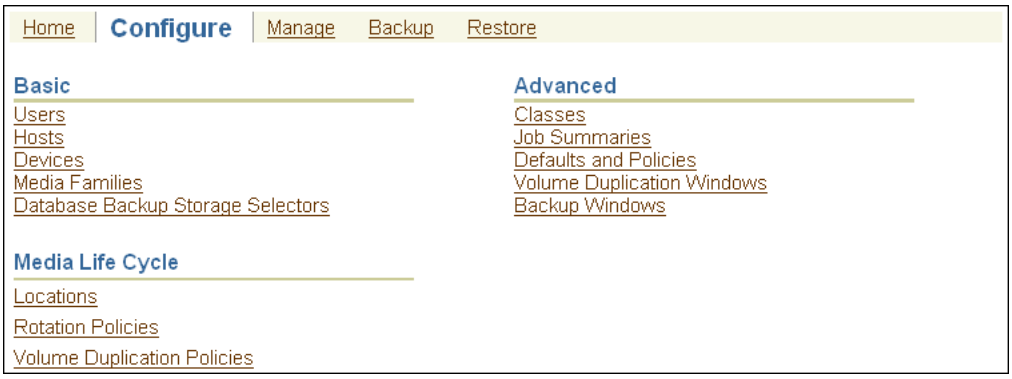
- **About**

  This link displays information about the Oracle Secure Backup software, including release date, system information, **administrative server** name, and IP address.

Web Tool Configure Page

Click the **Configure** tab from the menu bar to display configuration options. Figure 4–3 shows an example of the Configure page.

Figure 4–3 Oracle Secure Backup Configure Page



The Configure page is divided into basic and advanced sections. The basic section contains the following links:

- **Users**

  Click this link to configure one or more user accounts for logging into and employing Oracle Secure Backup.

- **Hosts**

  Click this link to configure one or more hosts. A host is a computer that participates in the Oracle Secure Backup **administrative domain**.

- **Devices**

  Click this link to configure a **tape device** for use with Oracle Secure Backup. A tape device is a **tape drive** or **tape library** identified by a user-defined name.

- **Media Families**

  Click this link to configure media families. A **media family** is a named classification of backup volumes. A **volume** is a unit of media, such as an 8mm tape.

- **Database Backup Storage Selectors**

  Click this link to configure one or more tape devices and media families for use during Oracle database backup and restore operations.

The advanced section contains the following links:

- **Classes**

  Click this link to configure classes. A **class** defines a set of **rights** that are granted to a user. A class can apply to multiple users; however, each user is assigned to exactly one class.

- **Job Summaries**

  Click this link to create a **job summary schedule** for generation of job summaries for email distribution.

  A **job summary** is a generated text file report that tells you whether a backup operation was successful. Oracle Secure Backup can generate and email job summaries detailing the status of each **scheduled backup**.

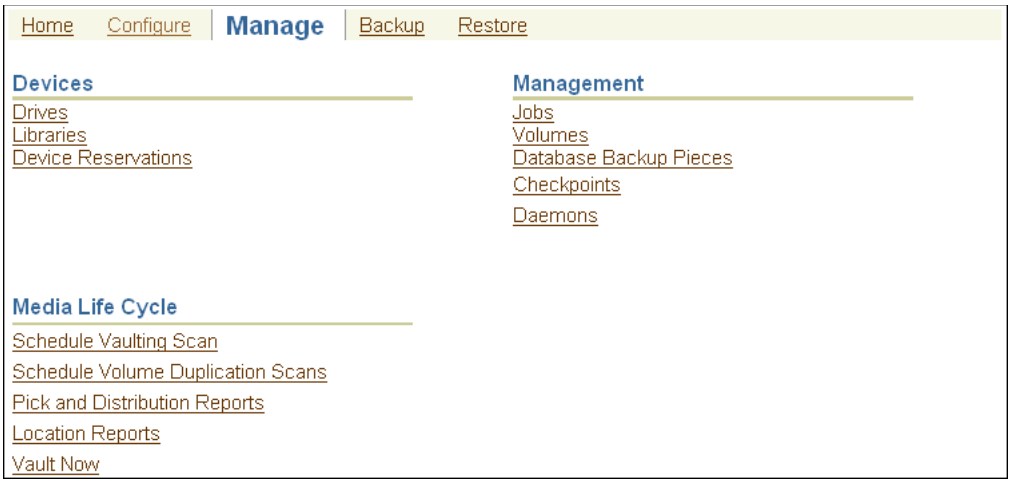
- **Defaults and Policies**

  Click this link to edit **defaults and policies**. Defaults and policies are sets of configuration data that control how Oracle Secure Backup runs throughout an administrative domain.

Web Tool Manage Page

Click the **Manage** tab to display management options. Figure 4–4 shows an example of the Manage page.

Figure 4–4 Oracle Secure Backup Manage Page



The Manage page is divided into two main sections. One is for Maintenance, and the other is for Devices and Media. The Devices and Media section includes the following links:

- **Drives**

  Click this link to determine the status of a **volume** or **tape device** or to mount or unmount a volume.

- **Libraries**

  Click this link to view and control libraries.

- **Device Reservations**

  Click this link to reserve and unreserve tape devices for private use.

The Maintenance section includes the following links:

- **Jobs**

  Click this link to manage jobs in an **administrative domain**. You can view the status of backup and restore jobs.

- **Volumes**

  Click this link to filter and then view all volumes in the **catalog**. You can filter the results to scale down your search. A volume is a unit of media, such as 8mm tape. A volume can contain multiple backup images.

- **Backup Images**

  Click this link to manage backup images. A backup image is the work product of a single backup operation.

- **Backup Sections**

  Click this link to view and remove backup sections. A **backup section** is that part of a backup image that occupies one physical volume.

- **Checkpoints**

  Click this link to list and delete checkpoints describing certain in-progress, failed, and completed **Network Data Management Protocol (NDMP)** backups.

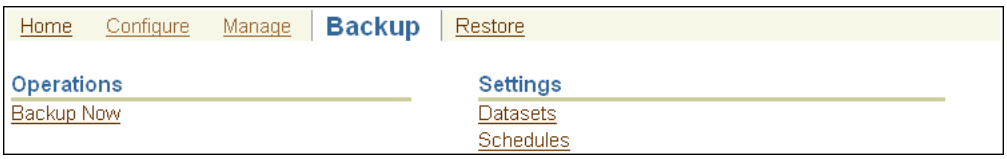
- **Daemons**

Click this link to manage **daemons** and control and view daemon properties.

Web Tool Backup Page

Click the **Backup** tab to display **backup image** options. Figure 4–5 shows a sample page.

Figure 4–5 Oracle Secure Backup Backup Page



The Backup page is divided into Operations and Settings sections. The Operations section contains the following link:

- **Backup Now**

  Click this link to perform one-time backups of data described by an existing **dataset file**.

The Settings section contains the following links:

- **Datasets**

  Click this link to configure dataset files. A dataset file describes the data to back up.

- **Schedules**

  Click this link to configure a **backup schedule**. The backup schedule describes the frequency with which a backup runs.

- **Backup Windows**

  Click this link to configure backup windows. A **backup window** is a time range for the execution of **scheduled backup** operations.

Web Tool Restore Page

Click the **Restore** tab to display restore options. Figure 4–6 shows a sample page.

Figure 4–6 Oracle Secure Backup Restore Page

The Restore page has a single Operations section with the following links:

- **Backup Catalog**

  Click this link to browse data associated with backup and restore operations.

- **Directly from Media**

  Click this link to perform raw restores, which require prior knowledge of the names of the file-system objects you want to restore. You must also know the volume IDs and the file numbers on which the volumes are stored.

Using obtool

obtool is the primary command-line interface to Oracle Secure Backup. The `obtool` executable is located in the bin subdirectory of the **Oracle Secure Backup home**. You can start `obtool` on any host in the **administrative domain**, log in to the domain as an **Oracle Secure Backup user**, and issue commands.

> **Note:** All examples in this section assume that the bin subdirectory of the Oracle Secure Backup home is in your PATH.

This section contains these topics:

- Displaying Help for Invoking obtool
- Starting obtool in Interactive Mode
- Running obtool Commands in Interactive Mode
- Executing obtool Commands in Noninteractive Mode
- Ending an obtool Session
- Starting obtool as a Specific User

> **See also:** *Oracle Secure Backup Reference* for a more detailed discussion of invoking `obtool` and for more information on obtar, which is mostly used internally by `obtool`

Displaying Help for Invoking obtool

Assuming that the bin subdirectory of the **Oracle Secure Backup home** is in your system path, you can obtain online help about **obtool** invocation options by running the following command at the operating system prompt:

```
% obtool help invocation
```

Starting obtool in Interactive Mode

Enter `obtool` at the command line to use **obtool** in interactive mode.

The first time you invoke `obtool`, you are required to establish your identity as an **Oracle Secure Backup user**. If you have not yet established a user identity, then `obtool` prompts you for a user name and password.

> **Note:** The installer for Oracle Secure Backup creates the `admin` user automatically, and prompts for a password. Use these credentials when you log in to Oracle Secure Backup for the first time after installation.
>
> The practice of supplying a password in clear text on a command line or in a command script is not recommended by Oracle. It is a security vulnerability. The recommended procedure is to have the user be prompted for the password.

Running obtool Commands in Interactive Mode

You can enter the commands described in *Oracle Secure Backup Reference* at the **obtool** prompt. For example, the `lshost` command displays information about the hosts in your **administrative domain**:

```
ob> lshost
brhost2          client                          (via OB)   in service
brhost3          mediaserver,client              (via OB)   in service
br_filer         client                          (via NDMP) in service
stadv07          admin,mediaserver,client        (via OB)   in service
```

Redirecting obtool Input from Text Files

You can use the < command in interactive mode to read text files containing multiple **obtool** commands. For example, you can create a file called my_script.txt with multiple `obtool` commands and redirect the `obtool` input to this script as follows:

```
ob> < /my_dir/my_script.txt
```

`obtool` runs the commands from the file and then returns to the `ob>` prompt for your next command.

Executing obtool Commands in Noninteractive Mode

You can run **obtool** in noninteractive mode from the Linux or UNIX shell or from the Windows command prompt with arguments that specify the command to run. `obtool` runs the specified command immediately and exits. Use the following syntax:

```
obtool  [ cl-option ]... command-name [ option ]... [ argument ]...
```

The following example runs the `lshost` command and then returns to the operating system prompt:

```
% obtool lshost
Output of command: lshost
brhost2          client                          (via OB)   in service
brhost3          mediaserver,client              (via OB)   in service
br_filer         client                          (via NDMP) in service
stadv07          admin,mediaserver,client        (via OB)   in service
%
```

Running Multiple Commands in Noninteractive Mode

You can run multiple commands in one invocation of **obtool** by separating the commands with a semicolon on the command line.

Note: Follow the quoting conventions of your host operating system shell or command line interpreter when entering a semicolon in the command line. For example, in a bash shell session, quote the semicolon as follows:

```
$ obtool lshost ';' lsdev
```

Redirecting Input in Noninteractive Mode

You can use the < command in noninteractive mode to read text files containing multiple **obtool** commands. For example, you can create a file called my_script.txt

with multiple `obtool` commands and redirect the `obtool` input to this script as follows:

```
% obtool < /my_dir/my_script.txt
```

`obtool` runs the commands from the file and then returns to the operating system prompt for your next command.

Ending an obtool Session

You can end an **obtool** session by using one of these commands:

- `exit`

  This command ends the `obtool` session, but a login token preserves your credentials, so that the next time you start `obtool` you are not prompted for a user name or password.

- `quit`

  This command is a synonym for `exit`.

- `logout`

  This command ends the `obtool` session and destroys the login token, so that you are prompted for credentials at the start of your next `obtool` session.

In the following example, login credentials are required for the first session, because the login token has expired. This first session is ended with an `exit` command, and a second session is started. No login credentials are required for this second session, because the login token was preserved. The second session is ended with a `logout` command, and a third session is started. The third session requires login credentials because the login token was destroyed by the `logout` command.

```
[cfoch@stbcs06-1 ~]$ obtool
Oracle Secure Backup 10.4.0.3.0
Warning: auto-login failed - login token has expired
login: admin
ob> exit
[cfoch@stbcs06-1 ~]$ obtool
ob> logout
[cfoch@stbcs06-1 ~]$ obtool
Oracle Secure Backup 10.4.0.3.0
login: admin
ob>
```

Starting obtool as a Specific User

You can force **obtool** to use different credentials when starting, destroying any existing login token. To do so, use the `-u` option with `obtool`, specifying the name of the user for the session. For example:

```
[root@osblin1 ~]# obtool -u admin
Password:
ob>
```

Note: The practice of supplying a password in clear text on a command line or in a command script is not recommended by Oracle. It is a security vulnerability. The recommended procedure is to have the user be prompted for the password.

5

Configuring and Managing the Administrative Domain

This chapter explains the basic steps involved in setting up an Oracle Secure Backup **administrative domain** after initial installation of the product on all of your hosts. Some steps, such as "Adding a Host to the Administrative Domain" on page 5-3, are also useful when managing an existing administrative domain.

This chapter contains the following sections:

Administrative Domain Configuration Overview

This section describes the steps involved in configuring an Oracle Secure Backup administrative domain. It assumes you have installed the Oracle Secure Backup software on each host in the domain, as described in Chapter 2, "Installing Oracle Secure Backup on Linux or UNIX" or Chapter 3, "Installing Oracle Secure Backup on Windows".

These instructions explain how to configure the administrative domain with host and **tape device** information using the Oracle Secure Backup **Web tool**. You can perform the same tasks using the **obtool** command-line interface to Oracle Secure Backup.

The instructions set up administrative domain security in a default security configuration that should be adequate for most users. Further configuration of users, user classes, security options, and the Oracle Secure Backup media management layer for use with **Recovery Manager (RMAN)** in backing up Oracle databases might be required in some cases. For details, see *Oracle Secure Backup Administrator's Guide*.

Administrative Domain Configuration Steps: Outline

The required steps to configure Oracle Secure Backup after installation are as follows:

1. Use your Web browser to connect to the Oracle Secure Backup Web tool running on the **administrative server** as the `admin` user. "Using the Oracle Secure Backup Web Tool" on page 4-4 describes this task.
2. For each host in your domain to be set up for the role of **media server**, perform the following steps:
   a. Add the host to the administrative domain. "Configuring the Administrative Domain with Hosts" on page 5-2 describes this task.

> **Note:** If the administrative server is also assigned the media server role, then it is already part of the administrative domain and does not need to be added.

b. Configure the administrative domain to include each tape device attached to this host. "Adding Tape Devices to an Administrative Domain" on page 5-11 describes this task.

3. For each host to be set up only for the **client** role, add the host to the administrative domain, as described in "Configuring the Administrative Domain with Hosts" on page 5-2.

   After configuring each client host, ping it to ensure that it is reachable.

4. Initial configuration is complete. Oracle Secure Backup is installed on all hosts, and all clients, media servers and tape devices are accessible by Oracle Secure Backup. Network communication among hosts in the administrative domain is configured with the default security configuration described in "Default Security Configuration" on page 6-15.

> **Note:** You must still identify files to be backed up, configure at least one **backup schedule**, and set up users, classes, and security policies. These tasks are described in the *Oracle Secure Backup Administrator's Guide*.

Configuring the Administrative Domain with Hosts

This section explains how to configure your administrative domain to add your hosts. This section contains these topics:

About Administrative Domain Host Configuration

The host configuration process makes the administrative server aware of a media server or client to be included in the administrative domain. You must perform this process for every host in the administrative domain, including each host running Oracle Secure Backup natively and each network-attached **storage device** managed by **Network Data Management Protocol (NDMP)**.

For any host to be added to the administrative domain, you must provide the following attributes:

- Host name
- IP address
- Assigned **roles**: client, media server or both
- Whether the host is in service or not in service at the moment

After adding a host to the administrative domain, Oracle recommends that you ping the host to confirm that it can be accessed by the administrative server.

> **See Also:** "Pinging a Host" on page 5-10

For hosts that use **NDMP access mode**, such as network-attached storage devices, you must configure the following additional attributes:

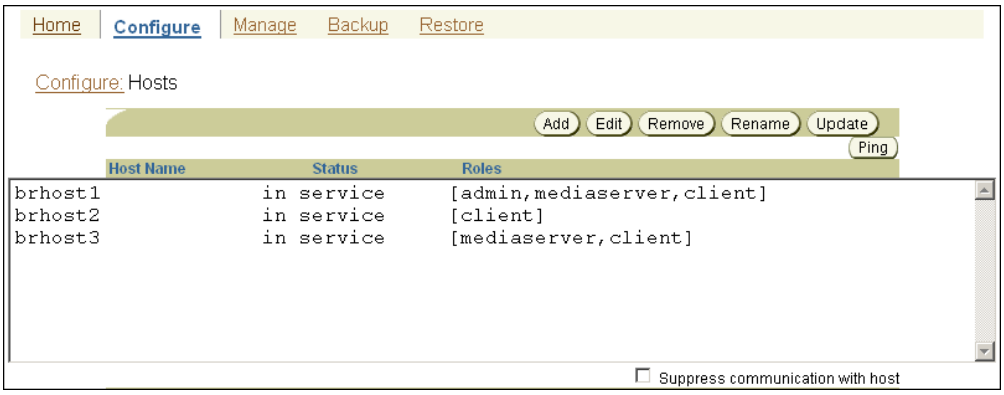
- NDMP authorization type
- NDMP password
- TCP port number for use with NDMP

> **See Also:** *Oracle Secure Backup Reference* for a complete account of host attributes

Viewing the Hosts in the Administrative Domain

In the Oracle Secure Backup Web tool, on the Configure page, click **Hosts** to display the Hosts page. The Hosts page lists the host name, configured host roles, and the current status of the host. Figure 5–1 shows a typical Hosts page.

Figure 5–1 Oracle Secure Backup Web Tool: Hosts Page



> **Note:** You can also view the current list of hosts with the **obtool** `lshost` command.

Adding a Host to the Administrative Domain

To add a host to an administrative domain:

1. From the Home page, click the **Configure** tab.
2. Click **Hosts** in the Basic section to display the Hosts page.

3. Click **Add** to add a host.

   The Oracle Secure Backup Web tool displays a form for entering configuration information about the host.

4. In the **Host** field, enter the unique name of the host in the Oracle Secure Backup administrative domain.

   In most cases, this name is the host name resolvable to an IP address using the host name resolution system (such as DNS or NIS) on your network. However, you can assign a different host name purely for use with Oracle Secure Backup.

   The name you enter must start with an alphanumeric character. It can contain only letters, numerals, dashes, underscores, and periods. The maximum length of a host name is 127 characters.

5. You must enter a value in the **IP Interface name(s)** field in the following situations:

   - The name of this host cannot be resolved to an IP address using a mechanism such as DNS or NIS
   - The resolvable name of your host is different from the value entered in the **Host** field.
   - Your host has multiple IP interface names or IP addresses to use with Oracle Secure Backup

   If any of the preceding conditions apply to this host, then enter one or more IP interface names in this field. Valid values are either resolvable host names or IP addresses. Separate multiple values with a comma.

   For example, you can use `myhost.oracle.com` for a host name or `141.146.8.66` for an IP address.

   If a value is specified for this field, then Oracle Secure Backup tries the host names or IP addresses in the order specified when it must contact this host, rather than using the name specified in the **Host** field.

   Note: If some hosts should contact this host using a particular network interface, then you can use the **Preferred Network Interface (PNI)** capability to override this order for those hosts, after completing the initial configuration of the administrative domain. See "Configuring Preferred Network Interfaces (PNI)" on page 5-8 for details.

   ---

6. In the **Status** list, select one of these:

   - **in service**

     Select this option to indicate that the host is available to perform backup and restore operations.

   - **not in service**

     Select this option to indicate that the host is unavailable to perform backup and restore operations.

7. In the **Roles** list, select the roles for this host: **admin**, **client** or **mediaserver**.

8. In the Disable RDS field, select one of the following:

   - **yes**

Select this option to disable the use of Reliable Datagram Socket (RDS) over Infiniband for communication between the client and media server. The default protocol, TCP/IP, is used for communication.

- **no**

  Select this option to enable the use of Reliable Datagram Socket (RDS) over Infiniband for communication between the client and media server.

- **systemdefault**

  Select this option to specify that the administrative domain level setting, by using the operations policy `disablerds`, is used to decide of RDS is enabled for the host. For example, if you set `systemdefault` at the host level and the `disablerds` policy is set to `no`, the host uses RDS for data transfer.

  See Also: Appendix E, "Oracle Secure Backup and Reliable Datagram Socket (RDS)" for more information about RDS

9. In the **Access method** field, select one of these:

   - **OB**

     Select this option for Windows, Linux and UNIX hosts that have Oracle Secure Backup installed.

   - **NDMP**

     Select this option for devices that support NDMP without an Oracle Secure Backup installation, such as a network-attached storage device.

   Note: **OB access mode** is a synonym for **primary access mode**. See "Oracle Secure Backup Host Access Modes" on page 1-3 for a discussion of access modes.

10. In **Public and private key sizes**, select the size for the public/private key associated with the **identity certificate** for this host.

    For hosts using the **ob** access mode, skip to Step 17. For hosts such as **Network Attached Storage (NAS)** devices that must use **NDMP** mode, continue to Step 11. Steps 11 through 16 apply only to hosts in NDMP mode.

11. In the **NDMP authorization type** list, select an authorization type. The authorization type defines the way Oracle Secure Backup authenticates itself to the NDMP server. Typically, you should use the default setting.

    Your choices are the following:

    - **default**

      Select this option to use the value of the Authentication type for the NDMP policy.

    - **none**

      Select this option to attempt to use the NDMP server from Oracle Secure Backup and provide no authentication data. This technique is usually unsuccessful.

    - **negotiated**

Select this option to negotiate with the NDMP server to determine the best authentication mode to use.

- **text**

  Select this option to use unencrypted text to authenticate.

- **md5**

  Select this option to use the MD5 digest algorithm to authenticate.

  See Also: *Oracle Secure Backup Administrator's Guide* to learn about NDMP-related policies

12. In the **Username** field, enter the name used to authenticate Oracle Secure Backup to this NDMP server. If left blank, then Oracle Secure Backup uses the name in the NDMP policy.
13. In the **Password** list, select one of these options:

    - **Use default password**

      Select this option to use the default NDMP password.

    - **Use text password**

      Select this option to enter a password.

    - **Set to NULL**

      Check this to use a NULL password.

    The password is used to authenticate Oracle Secure Backup to this NDMP server.
14. In the **Backup type** field, enter an NDMP backup type. A backup type is the name of a backup method supported by the NDMP **data service** running on a host. Backup types are defined by each data service provider.
15. In the **Protocol Version** list, select **2**, **3**, **4**, or **as proposed by server**. See "Oracle Secure Backup Host Access Modes" on page 1-3 for details on NDMP protocol versions.
16. In the **Port** field, enter a port number. Typically, the TCP port (10000) in the NDMP policy is used. You can specify another port if this server uses a port other than the default.
17. If the host you are adding to the administrative domain is not currently accessible on the network, then select the **Suppress communication with host** option.
18. Click **OK** to save your changes.

Adding the Media Server Role to an Administrative Server

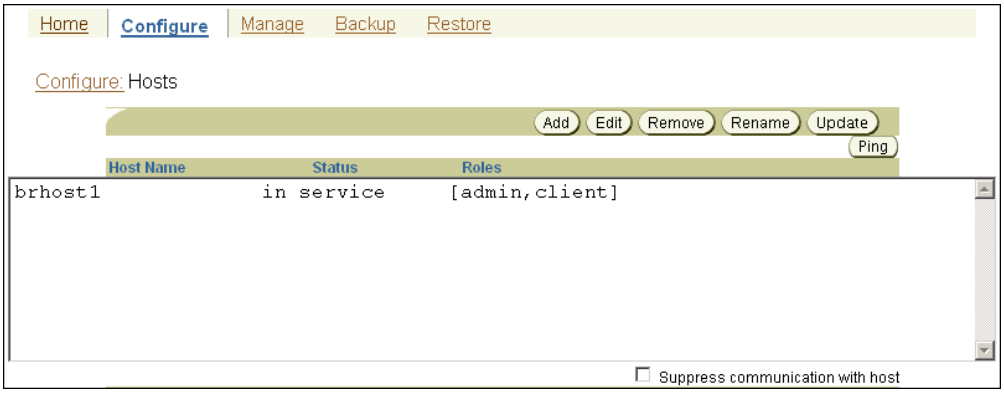
If you choose both the administrative server and media server roles when installing Oracle Secure Backup on a host, then that host is automatically part of the administrative domain. But it is not recognized as a media server until that role is explicitly granted to it using the `chhost` command in `obtool` or the Oracle Secure Backup Web tool.

See Also: *Oracle Secure Backup Reference* for complete syntax and semantics for the `chhost` command

Follow these steps to add the media server role to an administrative server using the Oracle Secure Backup Web tool:

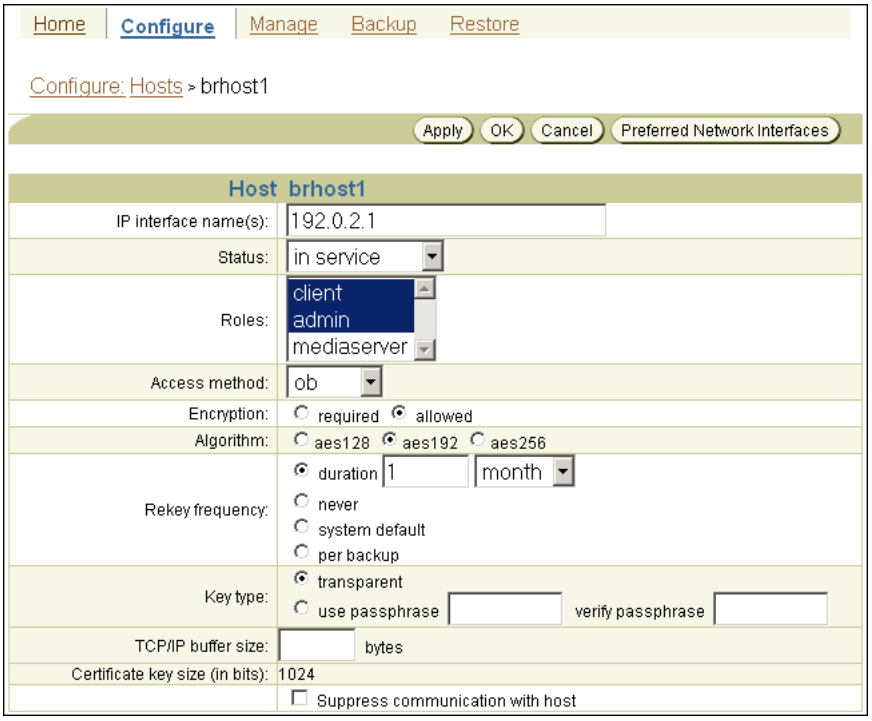
1. On the Configure page of the Oracle Secure Backup Web tool, click **Hosts**.

   The Configure: Hosts page appears.



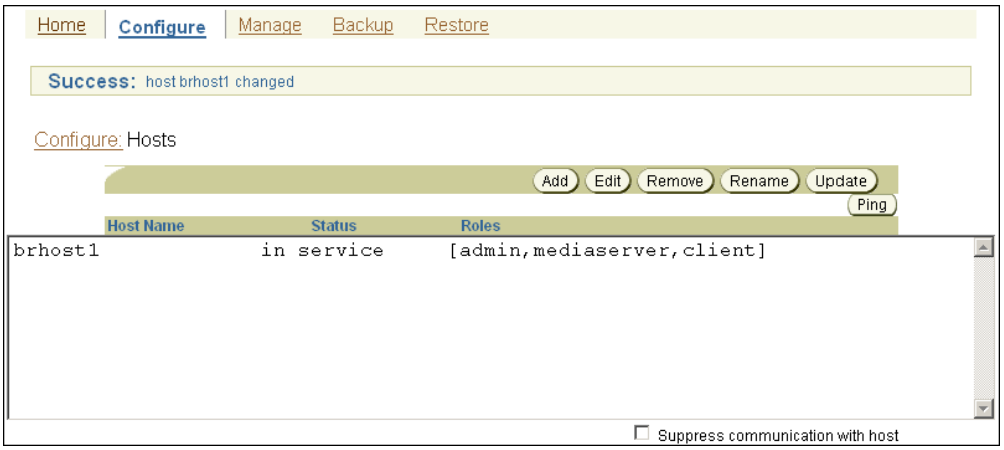
2. Select the administrative server and click **Edit**.

   The Configure: Hosts > *host_name* page appears.



3. In the Roles list, shift-click to add the media server role and then click **OK**.

   The Configure: Hosts page reappears with the media server role added to the administrative server host.



Adding Backup and Restore Environment Variables to an NDMP Host

Some NDMP hosts might require that you add backup and restore environment variables before they function with Oracle Secure Backup.

To add backup and restore variables:

1. In the field that appears next to the **Backup environment vars** or **Restore environment vars** field, enter a name-value pair.
2. Click **Add** to add the name-value pair as an environment variable.

   If an environment variable name or value includes spaces, then you must use quotes around the name or value to ensure correct processing of the name or value. For example, enter **A=B** or **"Name A"="Value B"** (if the name or value includes spaces).
3. Select an existing environment variable pair and click **Remove** to remove the pair.

Configuring Preferred Network Interfaces (PNI)

Multiple physical data paths can exist between a client, which contains primary storage to be backed up or restored, a media server, which controls at least one secondary storage device that writes and reads the backup media, and the administrative server. For example, a host might have multiple network interfaces connected to the network containing the hosts in the administrative domain. You can specify a PNI that identifies the network interface on a media server to use when transmitting backup or restore data to another specified host, or receiving data from that host.

To configure a Preferred Network Interface (PNI):

1. From the Configure page, select the host you want to configure and click **Edit**.

   The Configure Hosts > *host_name* page appears.
2. Click **Preferred Network Interfaces**.

   The Configure Hosts > *host_name* > Preferred Network Interface page appears.
3. Select an IP address or name from the **IP Address** list.

   This list shows each IP address or name by which this host can be referenced. Each is associated with a specific network interface. The IP address or name identifies the network interface that clients you select can use when communicating with the server.

4. Select one or more clients to use this IP address or DNS name from the **Host list** field.
5. Click **Add**.

   The Oracle Secure Backup Web tool displays the PNI in the **IP Address: Host List** field.

To remove a Preferred Network Interface (PNI):

1. In the **IP Address: Host List** field, select the name of the PNI to remove.
2. Click **Remove**.

PNI and Network Connection Types

When multiple network connections exist between a client and the media server, Oracle Secure Backup uses the following criteria to determine which connection type is used:

- If a PNI is configured, the network interface specified in the PNI is used to transfer backup and restore data between the client and media server.
- If a PNI is not configured, Oracle Secure Backup selects the connection type based on the order of precedence described in "Order of Precedence for Network Connection Types" on page 5-10.

For a particular connection type to be used, both the client and media server must support that connection type.

Network Load Balancing in Oracle Secure Backup

Network load balancing ensures that multiple network connections on a client are utilized optimally and no single connection carries the data load of all the concurrent backup and restore jobs. The transfer load of multiple backup and restore jobs is distributed across the network connections available on the client and media server. Load balancing is available starting with Oracle Secure Backup 10.4 and is supported for both file-system and Oracle Database backup and restore operations. Load balancing is turned off by default.

Note: Load balancing is not supported for NDMP clients.

Oracle Secure Backup sets up a data connection between the client and the media server over which the data transfer occurs. If a host contains more than one network interface of a particular type, Oracle Secure Backup uses all the available interfaces of that type for the data connections between the client and the media server. The type of network interface can be IPv4, IPv6, or RDS/RDMA (Reliable Datagram Socket over Remote Direct Memory Access) over Infiniband. Load balancing requires connectivity between the client and the media server on all the interfaces of the selected connection type.

Oracle Secure Backup selects a connection type only if both the client and the media server support that connection type. Therefore, if both the client and the media server support RDS/RDMA over Infiniband and the IPv6 connection types, then Oracle Secure Backup selects RDS/RDMA over Infiniband as the connection type.

If a Preferred Network Interface (PNI) is configured, then load balancing is disabled on the media server and PNI takes precedence. Load balancing will still be performed on the client.

Order of Precedence for Network Connection Types

When multiple network connections are available between a client and media server, Oracle Secure Backup decides which connection type to use based on the following order of precedence:

- RDS/RDMA over Infiniband
- IPv6
- IPv4 (includes TCP/IP over Infiniband)

Pinging a Host

You can use the Oracle Secure Backup ping operation to determine whether a host responds to requests from Oracle Secure Backup on each of its configured IP addresses.

Pinging a host attempts to establish a TCP connection to the host on each of the IP addresses you have configured for it. For hosts running Oracle Secure Backup, the connection occurs on TCP port 400. For hosts that use the NDMP access mode, connections occur through the configured NDMP TCP port, usually 10000.

Oracle Secure Backup reports the status of each connection attempt and immediately closes each connection that has been established successfully.

To ping a host:

1. From the Hosts page, select a host to ping.
2. Click **Ping**.

   A status line appears on the page with the results of the operation.

Viewing or Editing Host Properties

If you are having difficulties in configuration, then you might be required to view or edit the configuration of a host. To display or edit host properties:

1. From the Hosts page, select the name of the host whose properties require editing.

   Select the **Suppress communication with host** option to edit a host that is currently not accessible through the network.
2. Click **Edit**.

   The Oracle Secure Backup Web tool displays a page with details for the host you selected.
3. Make any desired changes to the host properties.
4. Click **OK** to save your changes.

Updating a Host

When you add or modify a host in an Oracle Secure Backup administrative domain, Oracle Secure Backup exchanges messages with that host to inform it of its changed state. If you select the **Suppress communication with host** option during an add or edit operation, however, then the host contains out-of-date configuration information. Use Update Host to send fresh state information to the host.

Updating is useful only for hosts running Oracle Secure Backup natively. Hosts accessed in NDMP mode, such as NAS devices, do not maintain any Oracle Secure Backup state data and therefore it is not necessary to update their state information.

To update a host:

1. From the Host page, select the name of the host to be updated.
2. Click **Update**.

Removing a Host

This section explains how to remove a host from an Oracle Secure Backup administrative domain. When you remove a host, Oracle Secure Backup destroys all information pertinent to that host, including:

- Configuration data
- Incremental backup state information
- Metadata in the backup **catalog** for this host
- Each device **attachment**
- PNI references

When you remove a host, Oracle Secure Backup contacts that host and directs it to delete the administrative domain membership information it maintains locally. You can suppress this communication if the host is no longer accessible.

To remove a host:

1. From the Hosts page, select the name of the host to remove.

   Check **Suppress communication with host** to remove a host that is not connected to the network.
2. Click **Remove**.

   Oracle Secure Backup prompts you to confirm the removal of the host.
3. Click **Yes** to remove the host or **No** to leave the host undisturbed.

   Oracle Secure Backup removes the host and returns you to the **Host** page.

Adding Tape Devices to an Administrative Domain

This section explains how to configure a tape drive or tape library for use with Oracle Secure Backup. This section contains these topics:

- Tape Device Names
- About Configuring Tape Drives and Libraries
- Displaying the Devices Page
- Configuring a Tape Library
- Configuring a Tape Drive
- Discovering Tape Devices Automatically on NDMP Hosts
- Configuring an NDMP Copy-Enabled Virtual Tape Library
- Adding a Tape Device Attachment
- Multiple Attachments for SAN-Attached Tape Devices
- Configuring Multihosted Device Objects

See Also: "Configuring the Solaris sgen Driver to Provide Oracle Secure Backup Attach Points" on page 2-19 to learn how to create attach points for tape devices on Solaris 10 systems

Tape Device Names

A tape device can be assigned a logical name by the host operating system (such as `nrst0a`), but it also can have a worldwide name, such as `nr.WWN[2:000:0090a5:0003f7]L1.a`. On some platforms, such as a **Fibre Channel tape drive** or tape library connected to a Network Appliance **filer**, the logical name might vary at each operating system restart. Oracle Secure Backup supports such tape devices, but they must be referred to by their worldwide name, which does not change across operating system restarts.

Any substring of the raw device name for the attachment that is the string `$WWN` is replaced with the value of the WWN each time the tape device is opened. For example a usable raw device name for a **Storage Area Network (SAN)** Network Appliance filer is `nr.$WWN.a`, specifying a no-rewind, best-compression tape device having the World Wide Name found in the device object.

The WWN is usually automatically discovered by the **device discovery** function in Oracle Secure Backup. However, you can enter it manually if necessary.

About Configuring Tape Drives and Libraries

This section explains how to configure a tape drive or tape library for use with Oracle Secure Backup. You can add a tape device in one of two ways:

- Manually

  A tape device connected to a media server on which Oracle Secure Backup is installed must be added to the administrative domain manually.

- Automatically discovery

  Oracle Secure Backup can automatically discover and configure each secondary storage device connected to certain types of NDMP servers, such as a Network Appliance filer.

Note: You must add the media server role to a host before adding any tape devices whose attachment point references that host. Oracle Secure Backup does not do this automatically.

For both tape drives and tape libraries, you can configure the following attributes:

- The name of the tape device
- The attachment, which is the description of a physical or logical connection of a tape device to a host
- Whether the tape device is in service

For tape drives, you can configure the following additional attributes:

- The tape library in which the tape drive is housed, if the tape drive is not standalone
- A **storage element** range that the tape device can use, if the tape drive is in a tape library

Note: Oracle Secure Backup identifies each tape drive within a tape library by its **data transfer element (DTE)** number. You must assign each tape device a DTE number if it is installed within a tape library. DTEs are numbered `1` through *n*. See the description of the `--dte` option to the `mkdev` command in *Oracle Secure Backup Reference* for more details on data transfer element numbers.

For tape libraries, you can configure the following additional attributes:

- Whether automatic cleaning is enabled
- The duration of a cleaning interval
- Whether a **barcode** reader is present

See Also: *Oracle Secure Backup Reference* for a complete account of tape device attributes.

To configure your administrative domain to include tape devices:

1. Disable any system software that scans and opens arbitrary SCSI targets before configuring Oracle Secure Backup tape devices.

   If Oracle Secure Backup has to contend with other system software (such as monitoring software) for access to tape libraries and tape drives, then unexpected behavior can result.

2. Configure tape libraries locally attached to your media servers, as described in "Configuring a Tape Library" on page 5-15.

   Configure tape drives locally attached to your media servers, as described in "Configuring a Tape Drive" on page 5-17

3. Configure tape devices that are network-accessible but are not locally attached.

   You must decide which media servers should control the tape devices and, for each media server, specify an attachment between the media server and the tape device. The procedure is identical to configuring a tape device attached locally to a media server.

4. Perform automatic device discovery to add every tape device attached to hosts that use NDMP access mode, such as NAS filers.

   "Discovering Tape Devices Automatically on NDMP Hosts" on page 5-20 describes this task.

5. Inventory each tape library, and then list its volumes.

   Each **volume** in a tape library should show either a barcode or the status unlabeled. If a library shows a slot as occupied, then this slot is in an invalid state.

Updating a Tape Device Inventory

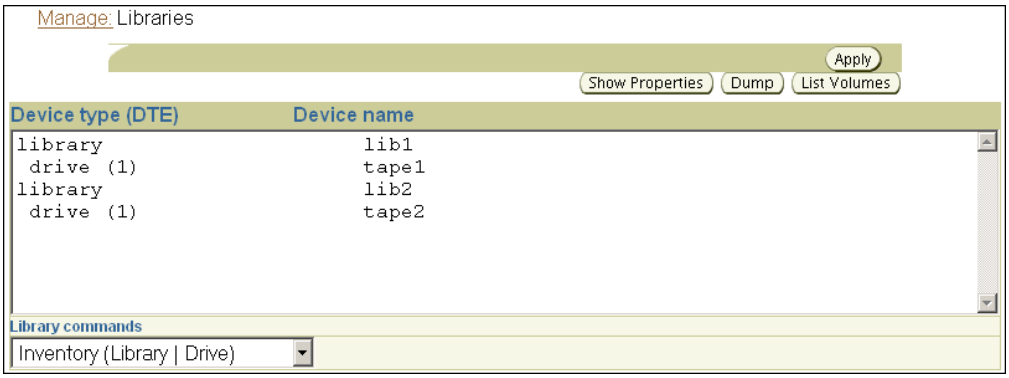
To update a tape library or tape drive inventory using the Oracle Secure Backup Web tool:

1. From the Oracle Secure Backup Web tool Home page, click **Manage**.

   The Manage page appears.

2. In the Devices section, click **Libraries**.

   The Manage: Libraries page appears.



3. Select the tape drive or tape library you want to inventory in the **Devices** table.
4. Select Inventory (Library | Drive) in the **Library commands** list.

   In this example, `lib1` is selected.
5. Click **Apply**.

   The Manage: Libraries page appears.
6. Ensure that the **Library** list is set to the device you want to inventory.
7. Select the **Force** option.

   Instead of reading from its cache, the tape library updates the inventory by physically scanning all tape library elements.
8. Click **OK**.

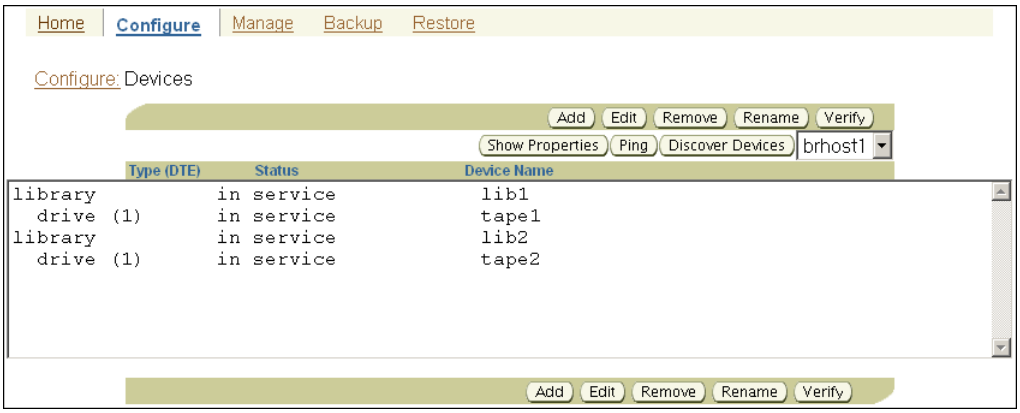
   When the inventory is complete, the Manage: Libraries page reappears and displays a success message.

   To see the results of the inventory, select the tape drive or tape library again and click **List Volumes**.

Displaying the Devices Page

The Devices page, illustrated in Figure 5–2, lists each tape library and tape drive that is currently in the administrative domain. The page lists the type, status, and name of every tape device.

Figure 5–2 Devices Page



Configuring a Tape Library

This section explains how to configure a tape library for use with Oracle Secure Backup.

To configure a tape library:

1. Disable any system software that scans and opens arbitrary SCSI targets before adding a tape device to an administrative domain. If Oracle Secure Backup has to contend with other system software (such as monitoring software) for access to a tape library or tape drive, then unexpected behavior can result.
2. From the Home page, click the **Configure** tab.
3. Click **Devices** in the Basic section to display the Devices page.
4. Click **Add** to add a tape device.
5. In the **Device** field, enter a name for the tape device.

   The name must start with an alphanumeric character. It can only contain letters, numerals, dashes, underscores, or periods. It can contain at most 127 characters.

   The tape device name is of your choosing. It must be unique among all Oracle Secure Backup device names. It is unrelated to any other name used in your computing environment or the Oracle Secure Backup administrative domain.
6. In the **Type** list, select **library**.
7. In the **Status** list, select one of these options:
   - **in service**

     Select this option to indicate that the tape device is available to perform Oracle Secure Backup backup and restore operations.
   - **not in service**

     Select this option to indicate that the tape device is unavailable to perform backup or restore operations.
   - **auto not in service**

     This option indicates that the tape device is unavailable to perform backup or restore operation and is set automatically for a failed operation.
8. In the **Debug mode** list, select **yes** or **no**. The default is **yes**.
9. In the **World Wide Name** field, enter a worldwide name for the tape device, if required.

   See Also: "Tape Device Names" on page 5-12 for more information on World Wide Names
10. In the **Barcode reader** list, select one of these options to indicate whether a barcode reader is present:
    - **yes**

      Select this option to indicate that the tape library has a barcode reader.
    - **no**

      Select this option to indicate that the tape library does not have a barcode reader.
    - **default**

Select this option to indicate that Oracle Secure Backup should automatically determine the barcode reader using information reported by either the tape library, the external device file, or both.

11. In the **Barcode required** list, select **yes** or **no**. If you specify **yes**, then Oracle Secure Backup refuses to use any tape that lacks a readable barcode.

    By default, Oracle Secure Backup does not discriminate between tapes with readable barcodes and those without. This policy ensures that Oracle Secure Backup can always solicit a tape needed for a restore operation by using either the barcode or the **volume ID**.

12. Set whether the tape library should use automatic cleaning.

    See Also: "Configuring Automatic Tape Cleaning for a Library" on page 5-17

13. In the **Unload required** list, select **yes** or **no** to specify if an unload operation is required before moving a tape from a tape drive to a storage element.

    The default value is **no**.

14. Select an ejection type. Your choices are:

    - Automatic

      Whenever a volume becomes eligible to be ejected from the tape library, Oracle Secure Backup moves that volume to an export element and notifies the backup operator that it is available there. If no export elements are available, then Oracle Secure Backup requests operator assistance.

    - On demand

      Whenever a volume becomes eligible to be ejected from the tape library, Oracle Secure Backup marks the volume to that effect. A media movement job then waits for the operator to reply to the job. The operator replies to the job through the job transcript. When the operator replies to the job to continue, Oracle Secure Backup ejects all such volumes through export elements.

    - Manual

      No automation is used to eject volumes from the tape library. The backup operator determines which storage elements contain volumes ready to be ejected and manually removes them. This option can be useful when the tape library has no import/export slots.

15. Enter a value in the Minimum writable volumes field.

    When Oracle Secure Backup scans tape devices for volumes to be moved, it looks at this minimum writable volume threshold. If the minimum writable volume threshold is nonzero, and if the number of writable volumes in that tape library is less than this threshold, then Oracle Secure Backup creates a media movement job for the full volumes even if their rotation policy does not require it. When this happens, Oracle Secure Backup notes in the media movement job transcript that volumes have been moved early.

16. Click **OK** to save your changes.

    See Also: "Adding a Tape Device Attachment" on page 5-22

Configuring Automatic Tape Cleaning for a Library

Oracle Secure Backup can automatically clean each tape drive in a tape library. A cleaning cycle is initiated either when a tape drive reports that it needs cleaning or when a specified usage time has elapsed.

Oracle Secure Backup checks for cleaning requirements when a cartridge is either loaded into or unloaded from a tape drive. If at that time a cleaning is required, then Oracle Secure Backup loads a cleaning cartridge, waits for the cleaning cycle to complete, replaces the cleaning cartridge in its original storage element, and continues with the requested load or unload.

To configure automatic cleaning for a tape library:

1. In the **Auto clean** list, select **yes** to enable automatic tape drive cleaning or **no** to disable it. You can also manually request that a cleaning be performed whenever a tape drive is not in use.

   Note: Not all tape drives can report that cleaning is required. For those tape drives, you must define a cleaning interval.

   In the **Clean interval (duration)** field, enter a value and then select the cleaning frequency from the adjacent list. This interval is the amount of time a tape drive is used before a cleaning cycle is initiated. If automatic tape drive cleaning is enabled, then this duration indicates the interval between cleaning cycles.

2. In the **Clean using emptiest** field, select one of these options:

   - **yes**

     Select this option to specify the emptiest cleaning tape, which causes cleaning tapes to "round robin" as cleanings are required.

   - **no**

     Select this option use the fullest cleaning tape, which causes each cleaning tape to be used until it fills, then the next cleaning tape fills, and so on.

   If there are multiple cleaning tapes in a tape library, then Oracle Secure Backup must decide which to use. If you do not otherwise specify, then Oracle Secure Backup chooses the cleaning tape with the fewest number of cleaning cycles remaining.

3. Click **OK** to save your changes.

See Also: "Adding a Tape Device Attachment" on page 5-22

Configuring a Tape Drive

This section explains how to configure a tape drive for use with Oracle Secure Backup. If the tape drive you want to configure is attached to a tape library, then you must configure the tape library first, as described in "Configuring a Tape Library" on page 5-15.

To configure tape drives for use with Oracle Secure Backup:

1. Disable any system software that scans and opens arbitrary SCSI targets before adding a tape device to an administrative domain. If Oracle Secure Backup has to contend with other system software (such as monitoring software) for access to tape libraries and tape drives, then unexpected behavior can result.
2. From the Home page, click the **Configure** tab.

3. Click **Devices** in the Basic section to display the Devices page.
4. Click **Add** to add a tape device.
5. In the **Device** field, enter a name for the tape device.

   The name must start with an alphanumeric character. It can only contain letters, numerals, dashes, underscores, or periods. It can contain at most 127 characters.

   The tape device name is of your choosing. It must be unique among all Oracle Secure Backup device names. It is unrelated to any other name used in your computing environment or the Oracle Secure Backup administrative domain.
6. In the **Serial number** field, enter the serial number of the tape drive.

   This step is not required. But if you do not enter a serial number, then Oracle Secure Backup reads and stores the tape drive serial number the first time it opens the tape drive.

   If the `checkserialnumbers` policy is enabled and you change the tape drive hardware, then you must enter the serial number of the tape drive before using it.

   See Also:

   - "Editing Device Properties" on page 5-26
   - *Oracle Secure Backup Reference* for more information on the `checkserialnumbers` policy
7. In the **Type** list, select **tape**.
8. In the **Status** list, select one of these options:
   - **in service**

     Select this option to indicate that the tape device is available to perform Oracle Secure Backup backup and restore operations.
   - **not in service**

     Select this option to indicate that the tape device is unavailable to perform backup or restore operations.
   - **auto not in service**

     This option indicates that the tape device is unavailable to perform backup or restore operation and is set automatically for a failed operation.
9. In the **Debug mode** list, select **yes** or **no**. The default is **yes**.
10. In the **World Wide Name** field, enter a worldwide name for the tape device, if required.

    See Also: "Tape Device Names" on page 5-12 for more information on World Wide Names
11. If the tape drive is located in a tape library, then select the tape library by name from the **Library** list.
12. In the **DTE** field, enter the **data transfer element (DTE)**.

    Note: This option is not available for standalone tape drives.

13. In the **Automount** field, select **yes** (default) or **no** to specify whether automount mode is on or off. Enable the automount mode if you want Oracle Secure Backup to mount tapes for backup and restore operations without **operator** intervention.
14. In the **Error rate** field, enter an **error rate** percentage or leave this field blank to accept the default setting. The default is **8**.

    The error rate is the ratio of restored write errors that occur during a **backup job** divided by the total number of blocks written, multiplied by 100. If the error rate for any backup is higher than this setting, then Oracle Secure Backup displays a warning message in the **backup transcript**.

    Oracle Secure Backup also issues a warning if it encounters a SCSI error when trying to read or reset the tape drive error counters. Some tape drives do not support the SCSI commands necessary to perform these operations. To avoid these warnings, error rate checking can be disabled by selecting **None**.
15. In the **Blocking factor** field, enter the **blocking factor** or leave this field blank to accept the default setting. The default is 128 bytes.

    The blocking factor value specifies how many 512-byte records to include in each block of data written to tape. The default value is 128, which means that Oracle Secure Backup writes 64K blocks to tape.

    > **See Also:** "Tape Drives" on page 1-5 for more information on blocking factors and maximum blocking factors
16. In the **Max Blocking factor** field, enter the maximum blocking factor.

    The largest value supported for the maximum blocking factor is 4096. This represents a maximum tape block size of 2MB.

    > **Note:** Device and operating system limitations might reduce this maximum block size.
17. In the **Drive usage** field, enter the amount of time the tape drive has been in use since it was last cleaned and then select the time unit from the adjacent list.
18. Leave the **Current tape** field empty during initial configuration. Update the tape drive inventory after configuration, as described in "Updating a Tape Device Inventory" on page 5-13.
19. In the **Use list** group, select one of these options to configure the use list:
    - **Storage element range or list**

      Select this option for a numeric range of storage element addresses. Enter a range in the field, for example, **1-20**.
    - **All**

      Select this option to specify all storage elements. For tape libraries with single tape drives, you can select this option to use all tapes. This is the default setting.
    - **None**

      Select this option to indicate that no storage elements have yet been specified. If you select **All** or **Storage element range or list**, then this option is no longer visible.

Oracle Secure Backup allows all tapes to be accessed by all tape drives. The use list enables you to divide the use of the tapes for tape libraries in which you are using multiple tape drives to perform backups. For example, you might want the tapes in half the storage elements to be available to the first tape drive, and those in the second half to be available to the second tape drive.

20. Click **OK** to save your changes.

Discovering Tape Devices Automatically on NDMP Hosts

Oracle Secure Backup can detect changes in tape device configuration for some types of hosts accessed by NDMP, such as a **filer**, and it can automatically update the administrative domain device configuration based on this information,.

Oracle Secure Backup detects and acts on these kinds of changes:

- Tape devices that were not previously part of the administrative domain are discovered. For each such tape device, Oracle Secure Backup creates a device with an internally-assigned name and configures a device attachment for it.
- If a previously configured tape device has an attachment, then Oracle Secure Backup adds an attachment to the existing device.
- If a previously configured tape device has lost an attachment, then Oracle Secure Backup deletes the attachment from the device.

Oracle Secure Backup detects tape devices that have multiple attachments by comparing the serial numbers for each tape device reported by the operating system. Oracle Secure Backup also determines whether any discovered tape device is accessible by its serial number. If the tape device is accessible by serial number, then Oracle Secure Backup configures each device attachment to reference the serial number instead of any logical name assigned by the operating system.

To discover tape devices attached to an NDMP host:

1. On the **Hosts** page select the name of the NDMP host in the list of hosts.
2. Click **Discover**.

   If changed tape devices are discovered, then the Oracle Secure Backup Web tool displays a message similar to the following:

   ```
   Info: beginning device discovery for host_name
   host_name_c0t0l0  (new library)
   WWN: [none]
   new attach-point on host_name, rawname c0t0l0
   host_name_c0t0l1  (new drive)
   WWN: [none]
   new attach-point on host_name, rawname c0t0l1
   host_name_c0t0l2  (new drive)
   WWN: [none]
   new attach-point on host_name, rawname c0t0l2
   ```

   If there are no changed tape devices to discover, then the Oracle Secure Backup Web tool displays a message similar to the following:

   ```
   Info: beginning device discovery for host_name.
   Info: no device configuration changes found for host_name
   ```

3. Click **OK** to return to the Devices page. The list of tape devices now includes the discovered tape devices.

Configuring an NDMP Copy-Enabled Virtual Tape Library

An NDMP copy-enabled virtual tape library (VTL) is a virtual tape library with an embedded NDMP server and multiple access paths. The embedded NDMP server allows offloading the I/O associated with volume duplication from the application running on the media server to the VTL.

An NDMP copy-enabled virtual tape library (VTL) must be represented in Oracle Secure Backup as a group of tape devices with multiple attach specifications. This ensures that the inventory data coming through the multiple access paths is identical.

Two Oracle Secure Backup host objects must be created to represent the VTL. One object must be associated with the media server to which the VTL is attached. The other host object must be associated with the VTL's embedded NDMP server. Both host objects must be assigned the media server role in Oracle Secure Backup.

One Oracle Secure Backup library device object with two attach specifications must be created for the virtual library. One access path is through the media server to which the VTL is attached. The other access path is through the embedded NDMP server.

An Oracle Secure Backup tape device object with two access paths must also be created for each virtual drive contained within the virtual library. As in the virtual library case, one access path is through the media server, and the other is through the embedded NDMP server.

One Oracle Secure Backup library device object with a single attach specification must be created for the physical library. The access path is through the VTL's embedded NDMP server. An Oracle Secure Backup tape device object with a single attach specification must also be created for each physical drive contained within the physical library. As in the physical library case, the access path is through the VTL's embedded NDMP server.

Note: Multiple media servers may be able to access the physical library and its drives if they are all connected to a shared SAN. In this case, the Oracle Secure Backup device objects for the physical library and its drives must be created with multiple attach points.

Here is an example of the `obtool` commands that would be used to configure an NDMP copy-enabled VTL. Many of the options that would be specified in a real environment have been omitted for clarity. Also, the device names shown are simply placeholders that may differ from the actual names in a real environment.

1. This command creates the Oracle Secure Backup host object associated with the media server to which the VTL is attached.

   ```
   mkhost --access ob --ip ipname osb_media_server
   ```

2. This command creates the Oracle Secure Backup host object associated with the embedded NDMP server contained within the VTL.

   ```
   mkhost --access ndmp --ip ipname ndmp_server
   ```

3. This command configures an Oracle Secure Backup device object that is associated with the virtual library *vlib*.

   ```
   mkdev --type library --class vtl
   --attach osb_media_server:/dev/obl0,ndmp_media_server:/dev/sg0 vlib
   ```

This library and its drives are accessible through the Oracle Secure Backup media server and the embedded NDMP server.

4. This command configures an Oracle Secure Backup device object that is associated with virtual tape drive *vdrive1*, which is contained in the virtual library *vlib*.

```
mkdev --type tape --library vlib --dte 1
--attach osb_media_server:/dev/obt0,ndmp_media_server:/dev/nst0 vdrive1
```

This command must be repeated for each tape drive in the virtual tape library.

5. This command configures an Oracle Secure Backup device object that is associated with the physical library *plib*.

```
mkdev --type library --attach ndmp_media_server:/dev/sg1 plib
```

This library and its drives are accessible only through the embedded NDMP server.

6. This command configures an Oracle Secure Backup device object that is associated with tape drive *pdrive1*, which is contained in the physical library *plib*.

```
mkdev --type tape --library plib --dte 1
--attach ndmp_media_server:/dev/nst1 pdrive1
```

See Also: *Oracle Secure Backup Administrator's Guide* for more information on NDMP copy-enabled virtual tape libraries

Adding a Tape Device Attachment

Oracle Secure Backup distinguishes between a tape device and a device attachment. A device attachment is the means by which that tape device is connected to a host. Each tape device can have one or more attachments, where each attachment describes a data path to the tape device from a host in the administrative domain.

An attachment is defined by the identity of the host to which the tape device is attached, and one of these names that represents the tape device on the host:

- Linux or UNIX **attach point** name
- Windows device name
- NAS device name

Note: For some older NAS devices, Oracle Secure Backup requires additional information to complete the attachment definition.

Before configuring a device attachment, refer to the description of the `mkdev` command in *Oracle Secure Backup Reference*. The description of the *aspec* placeholder describes the syntax and naming conventions for device attachments.

To configure a device attachment:

1. After adding or editing a device, click **Attachments**.
2. Select a host in the **Host** list.
3. In the **Raw device** field, enter the raw device name. This is the operating system's name for the device, such as a Linux or UNIX attach point or a Windows device file. For example, a tape library name might be `/dev/obl0` on Linux and `//./obl0` on Windows.

4. This step is required only for hosts running certain NDMP version 2 and 3 servers, such as Network Appliance Data ONTAP 5.1 or 5.2.
   a. In the **ST device** field, enter a device name.
   b. In the **ST target** field, enter a target number.
   c. In the **SCSI device** field, enter a SCSI device.
   d. In the **ST controller** field, enter a bus target number.
5. In the **ST lun** field, enter a **SCSI LUN** for the device.
6. Click **Add** to add the attachment.

Pinging a Device Attachment

You can ping a device attachment to determine whether the tape device is accessible to Oracle Secure Backup using that attachment. Pinging device attachments is a good way to test whether you set up the attachment properly.

When you ping a device, Oracle Secure Backup performs the following steps:

1. Establishes a logical connection to the device
2. Inquires about the device's identity data with the `SCSI INQUIRY` command
3. Closes the connection

If the attachment is remote from the host running the Oracle Secure Backup Web tool (or `obtool`), then Oracle Secure Backup establishes an NDMP session with the remote media server to effect this function.

To ping an attachment from the Attachments page:

1. Select the attachment to ping in the **host:raw device** field.
2. Click **Ping**.

   The Oracle Secure Backup Web tool opens a window that describes the status of the attachment.
3. Click **Close** to exit the page.

Displaying Device Attachment Properties

You can display device attachment properties from the Devices page.

To display attachment properties:

1. Select the name of the tape device whose attachment properties you want to view.
2. Click **Show Properties**.

   The Oracle Secure Backup Web tool displays device attachments and other properties for the tape device you selected.
3. Click **Close** to exit the page.

Multiple Attachments for SAN-Attached Tape Devices

A tape device attached to a SAN often has multiple attachments, one for each host with local access to the tape device through its Fibre Channel interface. A tape device attached to a SAN is also distinguished by a World Wide Name (WWN), an internal identifier that uniquely names the tape device on the SAN. Systems such as a Network Appliance filer permit access to tape devices attached to a SAN through their WWN.

Oracle Secure Backup includes a reference to the WWN in the device attachment's raw device name.

Tape devices such as certain Quantum and SpectraLogic tape libraries appear to be connected directly to an Ethernet LAN segment and accessed through NDMP. In fact, Oracle Secure Backup views these devices as having two discrete components:

- A host, which defines the IP address and which you configure through the Oracle Secure Backup Web tool Hosts page or the `mkhost` command
- A tape device, which has one attachment to the single-purpose host that serves as the front end for the tape device

Devices such as DinoStor TapeServer use a single host to service multiple tape devices.

For NDMP servers that run version 2, other data might be required to define SCSI parameters needed to access the tape device. These parameters are sent in an NDMP message called `NDMP_SCSI_SET_TARGET`. Oracle Secure Backup NDMP servers do not use this data or this message.

> **See Also:** The description of the `mkdev` command *`aspec`* placeholder in *Oracle Secure Backup Reference*, which describes the syntax and naming conventions for device attachments

Configuring Multihosted Device Objects

A **multihosted device**, also known as a **shared device**, is a tape library shared by multiple hosts within a single administrative domain. Shared devices are common in environments that deploy SAN or *i*SCSI-based tape equipment. These technologies give the user the flexibility to have multiple direct connections from hosts to tape devices, which enables all hosts to act as media servers.

When a device is shared by multiple hosts, you must create a single device object to ensure that the Oracle Secure Backup device reservation system to works correctly. You must then configure this device object to have a unique attach point that references each host sharing the device.

Table 5–1 shows the correct configuration of a single tape library and tape drive shared by two hosts: `host_a` and `host_b`. After the devices are configured, Oracle Secure Backup is aware of the devices and handles device reservation properly.

Table 5–1 Correct Configuration for Tape Library and Tape Drive

Tape Device Object	Attach Point 1	Attach Point 2
`SAN_library_1`	`host_a:/dev/sg1`	`host_b:/dev/sg5`
`SAN_tape_1`	`host_a:/dev/sg2`	`host_b:/dev/sg6`

If the device is configured as two separate device objects that point to the same physical device, then there is potential for contention. In this case, simultaneous backups to the these devices fail. Table 5–2 shows the *incorrect* configuration of a single tape library and tape drive shared by two hosts: `host_a` and `host_b`.

Table 5–2 Incorrect Configuration for Tape Library and Tape Drive

Tape Device Object	Attach Point
`SAN_library_1a`	`host_a:/dev/sg1`
`SAN_library_1a`	`host_b:/dev/sg5`

Table 5–2 (Cont.) Incorrect Configuration for Tape Library and Tape Drive

Tape Device Object	Attach Point
SAN_tape_1a	host_a:/dev/sg2
SAN_tape_1b	host_b:/dev/sg6

Creating Attach Points for Solaris 10 SCSI and Fibre Channel Devices

See "Configuring the Solaris sgen Driver to Provide Oracle Secure Backup Attach Points" on page 2-19.

Verifying and Configuring Added Tape Devices

This section explains how to verify that tape devices are reachable, display information about these devices, and configure serial number checking.

This section contains the following topics:

- Pinging a Tape Device
- Displaying Device Properties
- Editing Device Properties
- Verifying Tape Device Configuration
- Setting Serial Number Checking

Pinging a Tape Device

To determine whether a tape device is reachable by Oracle Secure Backup through any available attachment, ping the tape device. You should ping each tape device after it is configured or discovered, to verify that it is configured correctly.

To ping a tape device:

1. In the Devices page, select a tape device to ping.
2. Click the **Ping** button.

   The Oracle Secure Backup Web tool displays the status of the operation.

> **Note:** Pinging a tape library causes each service member tape drive in the tape library to be pinged as well.

Displaying Device Properties

The Oracle Secure Backup Web tool can display tape device properties including:

- Whether a tape device is in service
- Which host or hosts the tape device is connected to
- The tape device type

If a tape device is in service, then it Oracle Secure Backup can use it; if it is not in service, then Oracle Secure Backup cannot use it. When a tape device is taken out of service, no more backups are dispatched to it.

To display tape device properties:

1. In the Device page, select the name of the tape device whose properties you want to display.
2. Click **Show Properties**.

   The Oracle Secure Backup Web tool displays a page with the properties for the tape device you selected.

Editing Device Properties

If you make an error during installation, such as not configuring every attachment for a tape device or incorrectly configuring its properties, then you can edit its properties.

To edit the properties for an existing tape device:

1. From the Devices page, select the name of the tape device.
2. Click **Edit**.

   The Oracle Secure Backup Web tool displays a page with details for the tape device you selected.
3. Make any required changes.
4. Click **OK** to save your changes.

Verifying Tape Device Configuration

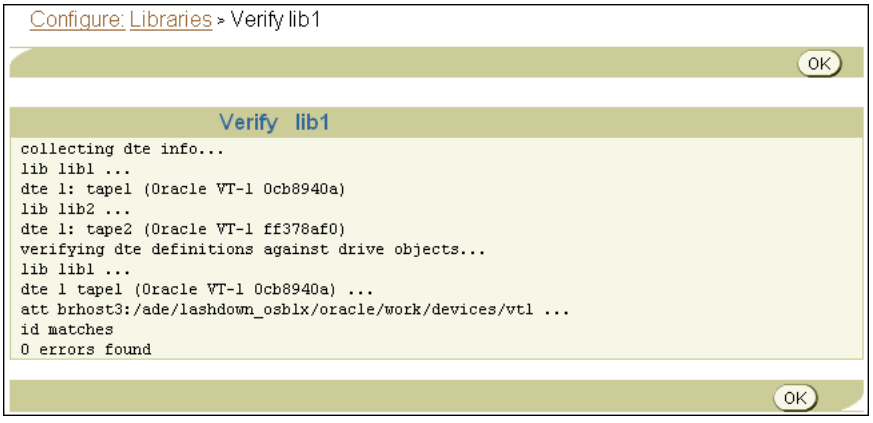
Oracle Secure Backup provides a method for checking for misconfigured tape and library devices:

1. From the Oracle Secure Backup Web tool home page, click **Configure**.

   The Configure page appears
2. In the Basic section click **Devices**.

   The Configure Devices page appears.
3. Select the drive whose configuration you want to check and click **Verify**.

   The Configure: Libraries > Verify *device_name* page appears.



In this example, library `lib1` is verified. No errors are found.

Setting Serial Number Checking

You can use the Oracle Secure Backup Web tool to enable or disable tape device serial number checking. If serial number checking is enabled, then whenever Oracle Secure

Backup opens a tape device, it checks the serial number of that device. If the tape device does not support serial number reporting, then Oracle Secure Backup simply opens the tape device. If the tape device does support serial number checking, then Oracle Secure Backup compares the reported serial number to the serial number stored in the device object. Three results are possible:

- There is no serial number in the device object.

  If Oracle Secure Backup has never opened this tape drive since the device was created or the serial number policy was enabled, then it cannot have stored a serial number in the device object. In this case, the serial number is stored in the device object, and the open succeeds.

- There is a serial number in the device object, and it matches the serial number just read from the device.

  In this case, Oracle Secure Backup opens the tape device.

- There is a serial number in the device object, and it does not match the serial number just read from the device.

  In this case, Oracle Secure Backup returns an error message and does not open the tape device.

> **Note:** Oracle Secure Backup also performs serial number checking as part of the `--geometry/-g` option to the `lsdev` command in `obtool`. This option causes an Inquiry command to be sent to the specified device, and `lsdev` displays its vendor, product ID, firmware version, and serial number.

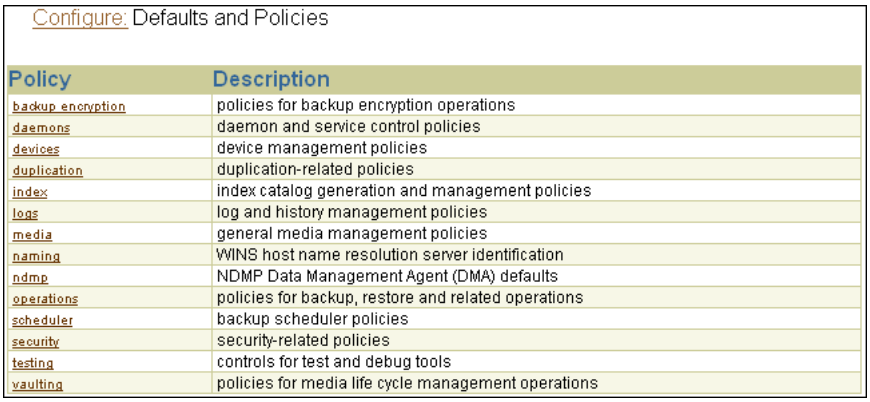
To enable or disable tape device serial number checking:

1. From the Oracle Secure Backup Web tool Home page, click **Configure**.

   The Configure page appears.

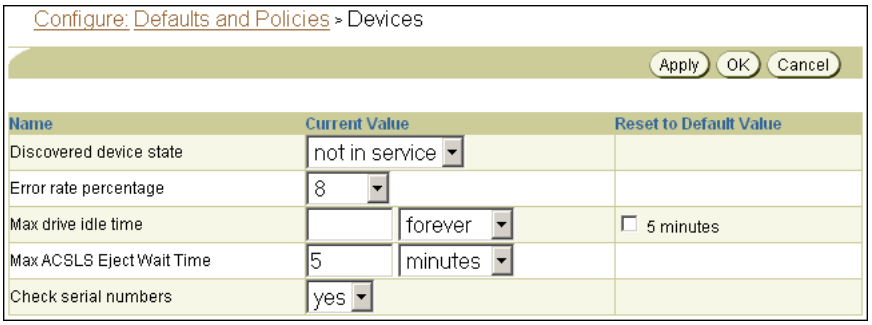
2. In the Advanced section, click **Defaults and Policies**.

   The Configure: Defaults and Policies page appears.



3. In the Policy column, click **devices**.

   The Configure: Defaults and Policies > Devices page appears.



4. Do one of the following:

   a. Select **Yes** from the **Check serial numbers** list to enable tape device serial number checking. This is the default setting.

   b. Select **No** from the **Check serial numbers** list to disable tape device serial number checking.

5. Click **OK**.

   The Configure: Defaults and Policies page appears with a success message.

6

Managing Security for Backup Networks

This chapter describes how to make your backup network more secure. Oracle Secure Backup is automatically configured for network security in your **administrative domain**, but you can enhance that basic level of security in several ways. Secure communications among the nodes of your administrative domain concerns the encryption of network traffic among your hosts. Secure communications is distinct from **Oracle Secure Backup user** and **roles** security concerns and security addressed by the encryption of backups to tape.

> **See Also:** *Oracle Secure Backup Administrator's Guide* for more information on users and roles management or **backup encryption**

This chapter contains these sections:

Backup Network Security Overview

An Oracle Secure Backup **administrative domain** is a network of hosts. Any such network has a level of vulnerability to malicious attacks. The task of the security administrator is to learn the types of possible attacks and techniques to guard against them. Your backup network must meet the following requirements to be both useful and secure:

- Software components must not expose the hosts they run on to attack.

  For example, **daemons** should be prevented from listening on a well-known port and performing arbitrary privileged operations.
- Data managed by the backup software must not be viewable, erasable, or modifiable by unauthorized users.
- Backup software must permit authorized users to perform these tasks.

Oracle Secure Backup meets these requirements in its default configuration. By default, all hosts that run Oracle Secure Backup must have their identity verified before they can join the administrative domain. A host within the domain uses an X.509 **certificate** for **host authentication**. After a **Secure Sockets Layer (SSL)** connection is established between hosts, control and data messages are encrypted when transmitted over the network. SSL protects the administrative domain from eavesdropping, message tampering or forgery, and replay attacks.

Network backup software such as Oracle Secure Backup is only one component of a secure backup network. Oracle Secure Backup can supplement but not replace the physical and network security provided by administrators.

Planning Security for an Administrative Domain

If security is of primary concern in your environment, then you might find it helpful to plan for network security in the following stages:

- Identifying Assets and Principals
- Identifying Your Backup Environment Type
- Choosing Secure Hosts for the Administrative and Media Servers
- Determining the Distribution Method of Host Identity Certificates

After completing these stages, you can proceed to the implementation phase as described in "Configuring Security for the Administrative Domain" on page 6-16.

Identifying Assets and Principals

The first step in planning security for an **administrative domain** is determining the assets and principals associated with the domain. The assets of the domain include:

- Database and file-system data requiring backup
- Metadata about the database and file-system data
- Passwords
- Identities
- Hosts and storage devices

Principals are users who either have access to the assets associated with the administrative domain or to a larger network that contains the domain. Principals include the following users:

- Backup administrators

  These Oracle Secure Backup users have administrative **rights** in the domain, access to the tapes containing backup data, and the rights required to perform backup and restore operations.

- Database administrators

  Each database administrator has complete access to his or her own database.

- Host owners

  Each host owner has complete access to its file system.

- System administrators

  These users might have access to the corporate network and to the hosts in the administrative domain (although not necessarily root access).

- Onlookers

  These users do not fall into any of the preceding categories of principals, but can access a larger network that contains the Oracle Secure Backup domain. Onlookers might own a host outside the domain.

The relationships between assets and principals partially determine the level of security in the Oracle Secure Backup administrative domain:

- In the highest level of security, the only principal with access to an asset is the owner. For example, only the owner of a **client** host can read or modify data from this host.
- In a medium level of security, the asset owner and the administrator of the domain both have access to the asset.
- In the lowest level of security, any principal can access any asset in the domain.

Identifying Your Backup Environment Type

After you have identified the assets and principals involved in your **administrative domain**, you can characterize the type of environment in which you are deploying the domain. The type of environment partially determines which security model to use.

The following criteria partially distinguish types of network environments:

- Scale

  The number of assets and principals associated with a domain plays an important role in domain security. A network that includes 1000 hosts and 2000 users has more points of entry for an attacker than a network of 5 hosts and 2 users.

- Sensitivity of data

  The sensitivity of data is measured by how dangerous it would be for the data to be accessed by a malicious user. For example, the home directory on a rank-and-file corporate employee's host is presumably less sensitive than a credit card company's subscriber data.

- Isolation of communication medium

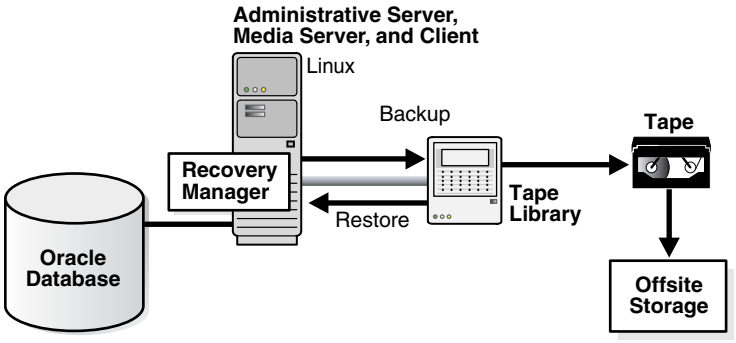
  The security of a network is contingent on the accessibility of network communications among hosts and devices in the domain. A private, corporate data center is more isolated in this sense than an entire corporate network.

The following sections describe types of network environments in which Oracle Secure Backup administrative domains are typically deployed. The sections also describe the security model typical for each environment.

Single System

The most basic **administrative domain** is illustrated in Figure 6–1. It consists of an **administrative server**, **media server**, and **client** on a single host.

Figure 6–1 Administrative Domain with One Host



This type of environment is small and isolated from the wider network. The data in this network type is probably on the low end of the sensitivity range. For example, the domain might consist of a server used to host personal Web sites within a corporate network.

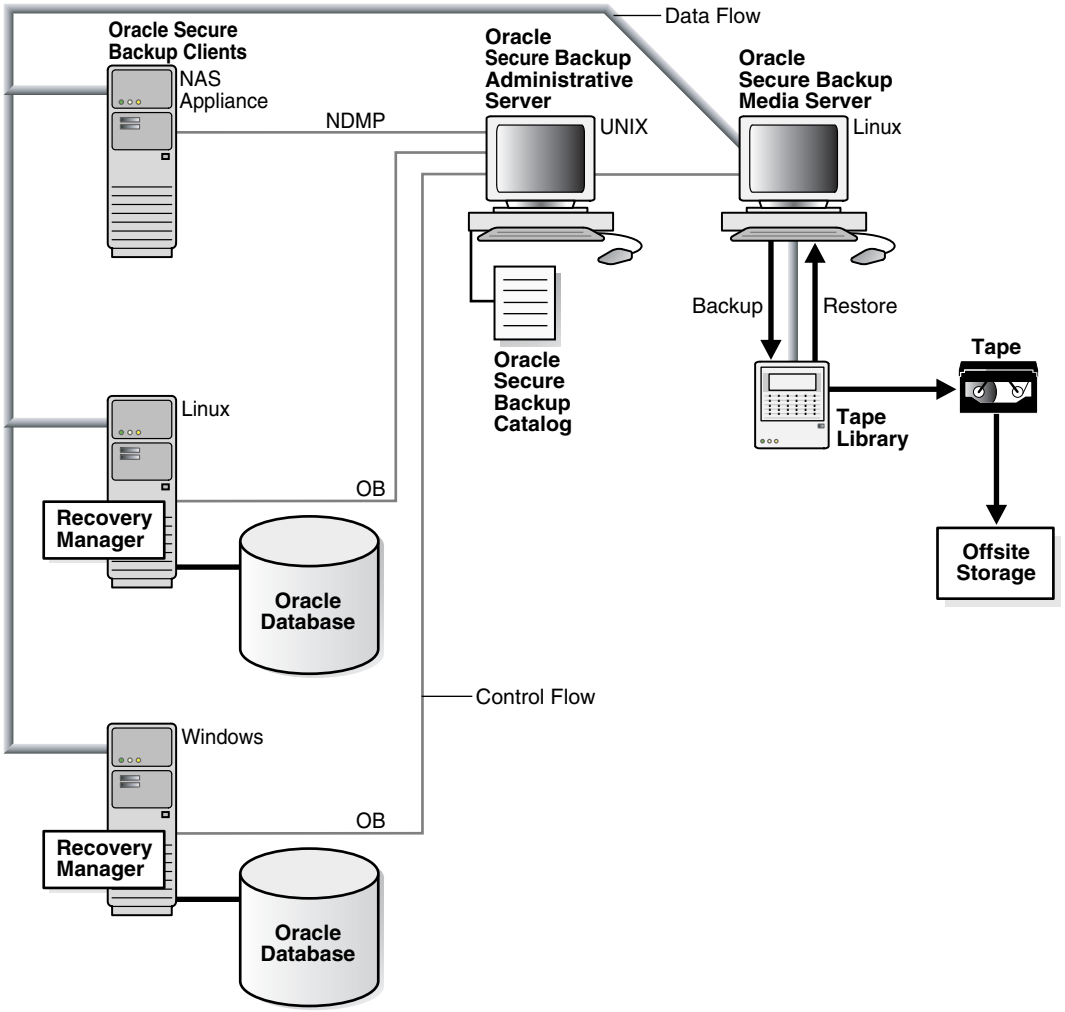
The assets include only a host and a **tape device**. The users probably include only the backup administrator and system administrator, who might be the same person. The backup administrator is the administrative user of the Oracle Secure Backup domain and is in charge of backups on the domain. The system administrator manages the hosts, tape devices, and networks used by the domain.

In this network type, the domain is fairly secure because it has one isolated host accessed by only a few trusted users. The administrator of the domain would probably not make security administration a primary concern, and the backup administrator could reasonably expect almost no overhead for maintaining and administering security in the Oracle Secure Backup domain.

Data Center

The **administrative domain** illustrated in Figure 6–2 is of medium size and is deployed in a secure environment such as a data center.

Figure 6–2 Administrative Domain with Multiple Hosts



The number of hosts, devices, and users in the administrative domain is much larger than in the single system network type, but it is still a small subset of the network at large. The data in this network type is probably on the high end of the sensitivity range. An example could be a network of hosts used to store confidential employee data. Network backups are conducted on a separate, secure, dedicated network.

The assets are physically secure computers in a dedicated network. The administrative domain could potentially include a dozen **media server** hosts that service the backups of a few hundred databases and file systems.

Principals include the following users:

- The backup administrator accesses the domain as an Oracle Secure Backup administrative user.
- The system administrator administers the computers, devices, and network.
- Database administrators can access their own databases and possibly have physical access to their database computers.
- Host administrators can access their file systems and possibly have physical access to their computers.

As with the single system network type, the administrative domain exists in a network environment that is secure. Administrators secure each host, **tape device**, and tapes by external means. Active attacks by a hacker are not likely. Administrators assume that security maintenance and administration for the domain requires almost no overhead. Backup and system administrators are primarily concerned with whether Oracle Secure Backup moves data between hosts efficiently.

Corporate Network

In this environment, multiple administrative domains, multiple **media server** hosts, and numerous **client** hosts exist in a corporate network.

The number of hosts, devices, and users in the administrative domains is extremely large. Data backed up includes both highly sensitive data such as human resources information and less sensitive data such as the home directories of low-level employees. Backups probably occur on the same corporate network used for e-mail, and Internet access. The corporate network is protected by a **firewall** from the broader Internet.

The assets include basically every piece of data and every computer in the corporation. Each administrative domain can have multiple users. Some host owners can have their own Oracle Secure Backup account to initiate a restore of their file systems or databases.

The security requirements for this backup environment are different from the single system and data center examples. Given the scope and distribution of the network, compromised client hosts are highly likely. For example, someone could steal a laptop used on a business trip. Malicious employees could illicitly log in to computers or run tcpdump or similar utilities to listen to network traffic.

The compromise of a client host must not compromise an entire administrative domain. A malicious user on a compromised computer must not be able to access data that was backed up by other users on other hosts. This user must also not be able to affect normal operation of the other hosts in the administrative domain.

Security administration and performance overhead is expected. Owners of sensitive assets must encrypt their backups, so physical access to backup media does not reveal the backup contents. The encryption and decryption must be performed on the client host itself, so sensitive data never leaves the host in unencrypted form.

Note: Oracle Secure Backup offers an optional and highly configurable **backup encryption** mechanism that ensures that data stored on tape is safe from prying eyes. Backup encryption is fully integrated with Oracle Secure Backup and is ready to use as soon as Oracle Secure Backup is installed. Backup encryption applies to both file-system data and **Recovery Manager (RMAN)** generated backups.

Choosing Secure Hosts for the Administrative and Media Servers

Your primary task when configuring security for your domain is providing physical and network security for your hosts and determining which hosts should perform the **administrative server** and **media server roles**.

When choosing administrative and media servers, remember that a host should only be an administrative or media server if it is protected by both physical and network security. For example, a host in a data center could be a candidate for an administrative server because it presumably belongs to a private, secured network accessible to a few trusted administrators.

Oracle Secure Backup cannot itself provide physical or network security for any host nor verify whether such security exists. For example, Oracle Secure Backup cannot stop malicious users from performing the following illicit activities:

- Physically compromising a host

  An attacker who gains physical access to a host can steal or destroy the primary or secondary storage. For example, a thief could break into an office and steal servers and tapes. Encryption can reduce some threats to data, but not all. An attacker who gains physical access to the administrative server compromises the entire **administrative domain**.

- Accessing the operating system of a host

  Suppose an onlooker steals a password by observing the owner of a **client** host entering his or her password. This malicious user could telnet to this host and delete, replace, or copy the data from primary storage. The most secure backup system in the world cannot protect data from attackers if they can access the data in its original location.

- Infiltrating or eavesdropping on the network

  Although backup software can in some instances communicate securely over insecure networks, it cannot always do so. Network security is an important part of a backup system, especially for communications based on **Network Data Management Protocol (NDMP)**.

- Deliberately misusing an Oracle Secure Backup identity

  If a person with Oracle Secure Backup administrator **rights** turns malicious, then he or she can wreak havoc on the administrative domain. For example, he or she could **overwrite** the file system on every host in the domain. No backup software can force a person always to behave in the best interests of your organization.

Determining the Distribution Method of Host Identity Certificates

After you have analyzed your backup environment and considered how to secure it, you can decide how each host in the domain obtains its **identity certificate**. Oracle Secure Backup uses **Secure Sockets Layer (SSL)** to establish a secure and trusted communication channel between domain hosts. Each host has an identity certificate signed by the **Certification Authority (CA)** that uniquely identifies this host within the domain. The identity certificate is required for authenticated SSL connections.

See Also:

- "Host Authentication and Communication" on page 6-9
- "Certification Authority" on page 6-10

The **administrative server** of the **administrative domain** is the CA for the domain. After you configure the administrative server, you can create each **media server** and **client** in the domain in either of the following modes:

- **automated certificate provisioning mode**

  In this case, no manual administration is required. When you configure the hosts, the CA issues identity certificates to the hosts over the network.

- **manual certificate provisioning mode**

  In this case, you must manually import the identity certificate for each host into its **wallet**.

Automated mode is easier to use but is vulnerable to unlikely man-in-the-middle attacks in which an attacker steals the name of a host just before you invite it to join the domain. This attacker could use the stolen host identity to join the domain illicitly. Manual mode is more difficult to use than automated mode, but is not vulnerable to the same kinds of attacks.

In manual mode, the administrative server does not transmit identity certificate responses to the host. Instead, you must carry a copy of the signed identity certificate on physical media to the host and then use the obcm utility to import the certificate into the wallet of the host. The obcm utility verifies that the certificate request in the wallet matches the signed identity certificate. A verification failure indicates that a rogue host likely attempted to masquerade as the host. You can reissue the `mkhost` command after the rogue host has been eliminated from the network.

See Also:

- "Managing Certificates with obcm" on page 6-21
- *Oracle Secure Backup Reference* for more information on the obcm utility

If you are considering manual certificate provisioning modes, then you must decide if the extra protection provided is worth the administrative overhead. Automated mode is safe in the single system and data center environments, because network communications are usually isolated.

Automated mode is also safe in the vast majority of corporate network cases. The corporate network is vulnerable to man-in-the-middle attacks only if attackers can insert themselves into the network between the administrative server and the host being added. This is the only place they can intercept network traffic and act as the man in the middle. This is difficult without the assistance of a rogue employee.

Manual certificate provisioning mode is recommended if the host being added is outside the corporate network, because communications with off-site hosts offer more interception and diversion opportunities.

Trusted Hosts

Starting with Oracle Secure Backup release 10.3 certain hosts in the **administrative domain** are assumed to have a higher level of security, and are treated as having an implicit level of trust. These hosts are the **administrative server** and each **media server**. These hosts are classified by Oracle Secure Backup as *trusted hosts*. Hosts configured with only the **client** role are classified as *non-trusted hosts*.

See Also: "Choosing Secure Hosts for the Administrative and Media Servers" on page 6-6

Many Oracle Secure Backup operations are reserved for use by trusted hosts, and fail if performed by a non-trusted host. These operations include:

- Use of **obtar** commands
- Direct access to physical devices and libraries
- Access to encryption keys

This policy provides an extra level of security against attacks that might originate from a compromised client system. For example, consider an Oracle Secure Backup administrative domain with host `admin` as the administrative server, host `media` as

the media server, and host `client` as the client. An **Oracle Secure Backup user** belonging to a **class** that has the `manage devices` class right attempts to run `lsvol -L` *`library_name`* in **obtool**. If the attempt is made on client, then it fails with an `illegal request from non-trusted host` error. The same command succeeds when attempted on `admin` or `media`.

You can turn off these trust checks by setting the Oracle Secure Backup security policy `trustedhosts` to `off`. This disables the constraints placed on non-trusted hosts.

> **Note:** Commands that originate from the Oracle Secure Backup **Web tool** are always routed to the administrative server for processing, and are not affected by the `trustedhosts` policy.

Host Authentication and Communication

By default, Oracle Secure Backup uses the **Secure Sockets Layer (SSL)** protocol to establish a secure communication channel between hosts in an **administrative domain**. Each host has an X.509 **certificate** known as an **identity certificate**. This identity certificate is signed by a **Certification Authority (CA)** and uniquely identifies this host within the administrative domain. The identity certificate is required for authenticated SSL connections.

> **Note:** Currently, the **Network Data Management Protocol (NDMP)** does not support an SSL connection to a **filer**.

This section contains these topics:

Identity Certificates and Public Key Cryptography

An **identity certificate** has both a body and a **digital signature**. The contents of a **certificate** include the following:

- A **public key**
- The identity of the host
- What the host is authorized to do

Every host in the domain, including the **administrative server**, has a **private key** known only to that host that is stored with the host's identity certificate. This private key corresponds to a public key that is made available to other hosts in the **administrative domain**.

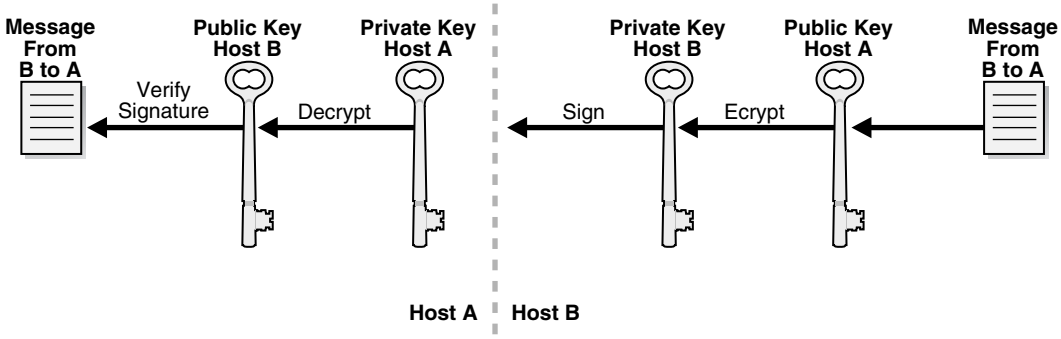
Any host in the domain can use a public key to send an encrypted message to another host. But only the host with the corresponding private key can decrypt the message. A host can use its private key to attach a digital signature to the message. The host

creates a digital signature by submitting the message as input to a **cryptographic hash function** and then encrypting the output hash with a private key.

The receiving host authenticates the digital signature by decrypting it with the sending host's public key. Afterwards, the receiving host decrypts the encrypted message with its private key, inputs the decrypted message to the same hash function used to create the signature, and then compares the output hash to the decrypted signature. If the two hashes match, then the message has not been tampered with.

Figure 6–3 illustrates how host B can encrypt and sign a message to host A, which can in turn decrypt the message and verify the signature.

Figure 6–3 Using Public and Private Keys to Encrypt and Sign Messages



Authenticated SSL Connections

For hosts to securely exchange control messages and backup data within the domain, they must first authenticate themselves to one another. Host connections are always two-way authenticated except for the initial host invitation to join a domain and communication with **Network Data Management Protocol (NDMP)** servers.

In two-way authentication, the hosts participate in a handshake process whereby they mutually decide on a cipher suite to use, exchange identity certificates, and validate that each other's **identity certificate** has been issued by a trusted **Certification Authority (CA)**. At the end of this process, a secure and trusted communication channel is established for the exchange of data.

The use of identity certificates and **Secure Sockets Layer (SSL)** prevents outside attackers from impersonating a **client** in the **administrative domain** and accessing backup data. For example, an outside attacker could not run an application on a non-domain host that sends messages to domain hosts that claim origin from a host within the domain.

Certification Authority

The **service daemon** (observiced) on the **administrative server** is the root **Certification Authority (CA)** of the **administrative domain**. The primary task of the CA is to issue and sign an **identity certificate** for each host in the administrative domain. The CA's signing **certificate**, which it issues to itself and then signs, gives the CA the authority to sign identity certificates for hosts in the domain. The relationship of trust requires that all hosts in the administrative domain can trust certificates issued by the CA.

Each host stores its own identity certificate and a **trusted certificate** (or set of certificates) that establishes a chain of trust to the CA. Like other hosts in the domain, the CA stores its identity certificate. The CA also maintains a signing certificate that authorizes the CA to sign the identity certificates for the other hosts in the domain.

Automated and Manual Certificate Provisioning Mode

Oracle Secure Backup provides automated and manual modes for initializing the security credentials for a **client** host that wants to join the domain. The automated mode is easy to use, but it has potential security vulnerabilities. The manual mode is harder to use, but it is less vulnerable to tampering.

In **automated certificate provisioning mode**, which is the default, adding a host to the domain is transparent. The host generates a **public key**/**private key** pair and then sends a **certificate** request, which includes the public key, to the **Certification Authority (CA)**. The CA issues the host an **identity certificate**, which it sends to the host along with any certificates required to establish a chain of trust to the CA.

The communication between the two hosts is over a secure but non-authenticated **Secure Sockets Layer (SSL)** connection. It is conceivable that a rogue host could insert itself into the network between the CA and the host, thereby masquerading as the legitimate host and illegally entering the domain.

In **manual certificate provisioning mode**, the CA does not automatically transmit certificate responses to the host. You must transfer the certificate as follows:

1. Use the obcm utility to export a signed certificate from the CA.
2. Use a secure mechanism such as a floppy disk or USB key chain drive to transfer a copy of the signed identity certificate from the CA to the host.
3. Use obcm on the host to import the transferred certificate into the host's **wallet**. The obcm utility verifies that the certificate request in the wallet matches the signed identity certificate.

You must balance security and usability to determine which certificate provisioning mode is best for your **administrative domain**.

Oracle Wallet

Oracle Secure Backup stores every **certificate** in an Oracle **wallet**. The wallet is represented on the operating system as a password-protected, encrypted file. Each host in the **administrative domain** has its own wallet in which it stores its **identity certificate**, **private key**, and at least one **trusted certificate**. Oracle Secure Backup does not share its wallets with other Oracle products.

Besides maintaining its password-protected wallet, each host in the domain maintains an **obfuscated wallet**. This version of the wallet does not require a password. The obfuscated wallet, which is scrambled but not encrypted, enables the Oracle Secure Backup software to run without requiring a password during system startup.

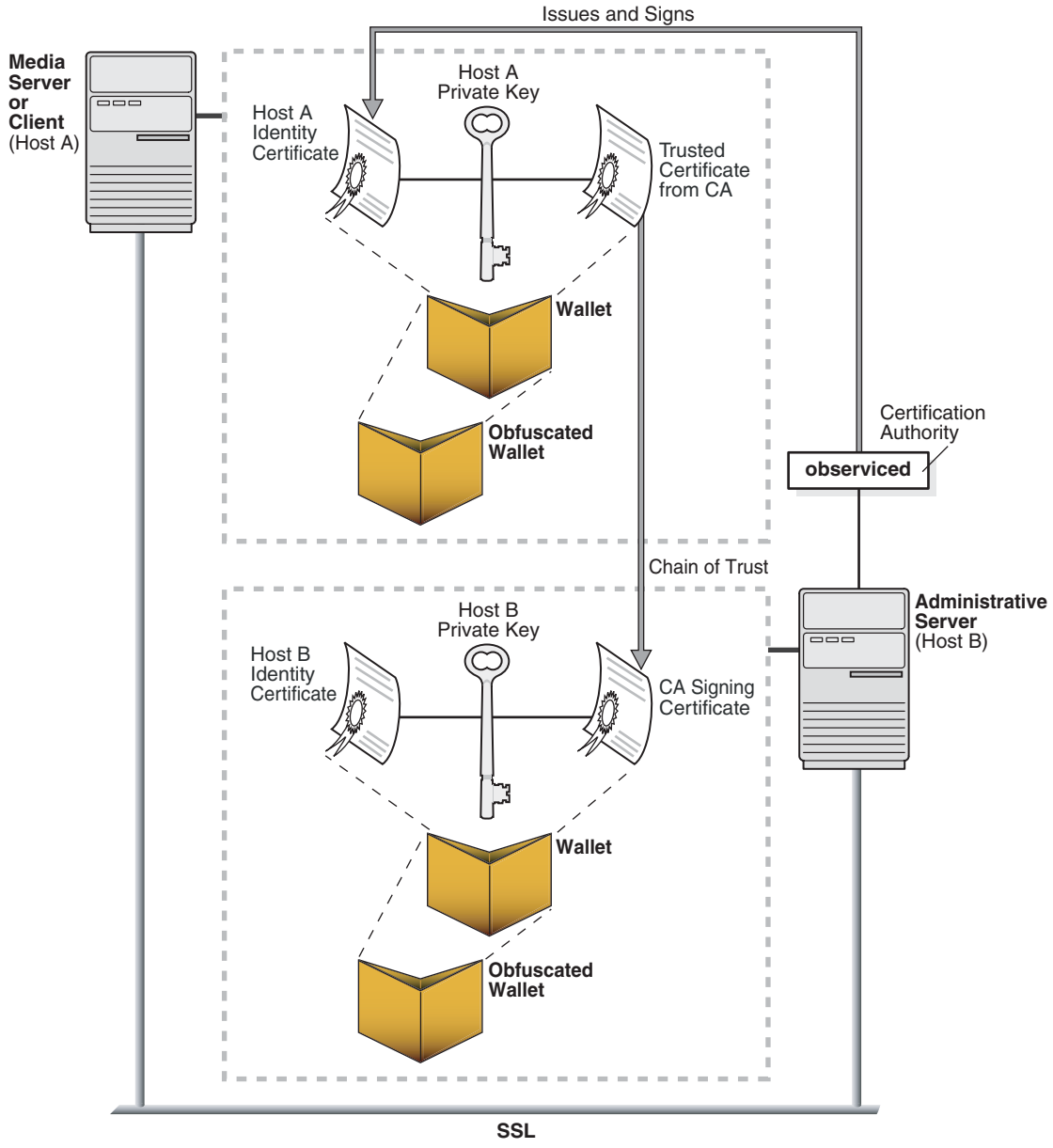
Note: To reduce risk of unauthorized access to obfuscated wallets, Oracle Secure Backup, by default, does not back them up. The obfuscated version of a wallet is named `cwallet.sso`. Oracle Secure Backup will backup a `cwallet.sso` file only if that file is included as part of an encrypted backup or the backup dataset includes an explicit path name of the `cwallet.sso` file. For more information about including an encrypted wallet in the dataset, see *Oracle Secure Backup Reference*.

By default, the wallet is located in `/usr/etc/ob/wallet` on Linux and UNIX and `C:\Program Files\Oracle\Backup\db\wallet` on Windows.

The password for the password-protected wallet is generated by Oracle Secure Backup and not made available to the user. The password-protected wallet is not usually used after the security credentials for the host have been established, because the Oracle Secure Backup **daemons** use the obfuscated wallet.

Figure 6–4 illustrates the relationship between the certificate authority and other hosts in the domain.

Figure 6–4 Oracle Wallets



Oracle Secure Backup Encryption Wallet

The **administrative server** has a second **wallet** that is used to store the master keys that encrypt secure data, such as the passwords for **Network Data Management Protocol (NDMP)** servers and the **backup encryption** key store. This wallet is separate from the wallet used for a host **identity certificate**. The key wallet is named `ewallet.p12` and is located in *`OSB_HOME`*`/admin/encryption/wallet`.

It is a best practice to use Oracle Secure Backup **catalog** recovery to back up the wallet.

If you do not use Oracle Secure Backup catalog recovery to back up the wallet, then Oracle recommends that the ewallet.p12 encryption wallet not be backed up on the same media as encrypted data. Encryption wallets are not excluded from backup operations automatically. You must use the `exclude` **dataset** statement to specify what files to skip during a backup:

```
exclude name *.p12
```

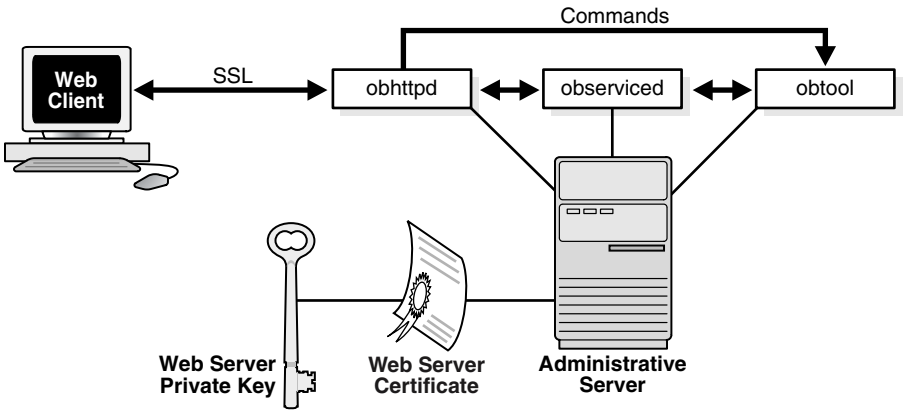
> **See Also:** *Oracle Secure Backup Administrator's Guide* for more information on dataset statements and catalog recovery

Web Server Authentication

The **Apache Web server** for the **administrative domain** runs on the **administrative server** as the obhttpd daemon. When you issue commands through the Oracle Secure Backup **Web tool**, obhttpd repackages them as **obtool** commands and passes them to an instance of `obtool` running on the administrative server.

The Web server requires a signed X.509 **certificate** and associated **public key**/**private key** pair to establish an **Secure Sockets Layer (SSL)** connection with a client Web browser. The X.509 certificate for the Web server is self-signed by the `installob` program when you install Oracle Secure Backup on the administrative server. Figure 6–5 shows the interaction between Web server and client.

Figure 6–5 Web Server Authentication



The Web server X.509 certificate and keys are not stored in the **wallet** used for **host authentication** in the Oracle Secure Backup administrative domain, but are stored in files in the `/apache/conf` subdirectory of the **Oracle Secure Backup home**. A single password protects the certificates and keys. This password is stored in encrypted form in the daemons file located in `/admin/config/default`. When the Web server starts, it obtains the password by using a mechanism specified in the Web server configuration file. This password is never transmitted over the network.

Revoking a Host Identity Certificate

Revoking a host **identity certificate** is an extreme measure that would only be performed if the backup administrator determined that the security of a computer in the Oracle Secure Backup **administrative domain** had been breached in some way.

You can revoke a host identity certificate with the `revhost` command in **obtool**.

> **See Also:** *Oracle Secure Backup Reference* for `revhost` syntax and semantics

If you revoke a host identity certificate, then none of the Oracle Secure Backup service **daemons** accept connections from that host. Revocation is not reversible. If you revoke a host identity certificate and then change your mind, then you must reinstall the Oracle Secure Backup software on the affected host.

Encryption of Data in Transit

Figure 1–2, "Oracle Secure Backup Administrative Domain with Multiple Hosts" on page 1-5 illustrates the control flow and data flow within an **administrative domain**. Control messages exchanged by hosts in the administrative domain are encrypted by **Secure Sockets Layer (SSL)**.

Data flow in the domain includes both file-system and database backup data. To understand how **backup encryption** affects data, it is helpful to distinguish between data at rest, which is backup data that resides on media such as disk or tape, and data in transit, which is backup data in the process of being transmitted over the network.

File-system backups and unencrypted RMAN backups on tape (data at rest) can be encrypted by Oracle Secure Backup. RMAN-encrypted backups made through the Oracle Secure Backup **SBT interface** are supported, but the encryption is provided by RMAN before the backup is provided to the SBT interface. The Oracle Secure Backup SBT interface is the only supported interface for making encrypted RMAN backups directly to tape.

> **See Also:** *Oracle Secure Backup Administrator's Guide* for more information on Oracle Secure Backup encryption

If you have selected RMAN or Oracle Secure Backup encryption, then Oracle Secure Backup does not apply additional encryption to data in transit within an administrative domain. If you have not selected either RMAN encryption or Oracle Secure Backup encryption, then backup data in transit, both file-system and database data, is not encrypted through SSL by default. To improve security, you can enable encryption for data in transit within the administrative domain with the `encryptdataintransit` security policy.

To enable **backup encryption** in the `encryptdataintransit` security policy:

1. Log in to **obtool** as a user with the `modify administrative domain's configuration` right.
2. Use the `setp` command to switch the `encryptdataintransit` policy to `no`, as shown in the following example:

```
ob> cdp security
ob> setp encryptdataintransit yes
```

> **See Also:** *Oracle Secure Backup Reference* for more information on the `encryptdataintransit` security policy

Suppose you want to back up data on **client** host client_host to a **tape drive** attached to **media server** media_server. Data encryption depends on what encryption options you choose and on what you are backing up, as shown in the following examples:

- Encrypted RMAN backup of a database on client_host.

RMAN encrypts the backup before it is provided to the SBT interface on client_host. Oracle Secure Backup transfers the RMAN-encrypted data over the network to media_server. Oracle Secure Backup does not apply additional encryption to the data as it passes over the network. After Oracle Secure Backup writes the data to tape, the data resides on tape in encrypted form.

- Unencrypted RMAN backup of a database on client_host.

  Oracle Secure Backup does not encrypt the data before transferring it over the network to media_server. After Oracle Secure Backup writes the data to tape, the data resides on tape in unencrypted form.

- Unencrypted RMAN backup of a database on client_host with `encryptdataintransit` set to `yes`.

  Oracle Secure Backup encrypts the data before transferring it over the network to media_server. The encrypted data is decrypted at media_server. After Oracle Secure Backup writes the data to tape, the data resides on tape in unencrypted form.

- Encrypted Oracle Secure Backup backup of the file system on client_host.

  Oracle Secure Backup transfers the encrypted backup data over the network to media_server. Oracle Secure Backup does not apply additional encryption to the data as it passes over the network. After Oracle Secure Backup writes the data to tape, the file-system data resides on tape in encrypted form.

- Unencrypted Oracle Secure Backup of the file system on client_host.

  Oracle Secure Backup does not encrypt the data before transferring it over the network to media_server. After Oracle Secure Backup writes the data to tape, the data resides on tape in unencrypted form.

- Unencrypted Oracle Secure Backup of the file system on client_host with `encryptdataintransit` set to `yes`.

  Oracle Secure Backup encrypts the data before transferring it over the network to media_server. The encrypted data is decrypted at media_server. After Oracle Secure Backup writes the data to tape, the data resides on tape in unencrypted form.

> **See Also:** *Oracle Database Backup and Recovery Advanced User's Guide* to learn about encryption of RMAN backups

Default Security Configuration

When you install Oracle Secure Backup on the **administrative server**, the installation program configures the administrative server as the **Certification Authority (CA)** automatically. By default, security for an **administrative domain** is configured as follows:

- **Secure Sockets Layer (SSL)** is used for **host authentication** and message integrity.
- The CA signs and issues the **identity certificate** for each domain host in **automated certificate provisioning mode**.
- The size of the **public key** and **private key** for every host in the domain is 1024 bits.
- Host communications within the domain are encrypted by SSL.

When you add hosts to the administrative domain, Oracle Secure Backup creates the **wallet**, keys, and certificates for each host when you create the hosts in **obtool** or the

Oracle Secure Backup **Web tool**. No additional intervention or configuration is required.

You can also change the default configuration in any of the following ways:

- Disable SSL for inter-host authentication and communication by setting the `securecomms` security policy
- Transmit identity certificates in **manual certificate provisioning mode**
- Set the key size for a host to a value greater or less than the default of 1024 bits
- Enable encryption for backup data in transit by setting the `encryptdataintransit` security policy

Configuring Security for the Administrative Domain

This section describes how to configure security for the **administrative domain**.

This section contains these topics:

Providing Certificates for Hosts in the Administrative Domain

Providing a **certificate** for each host in the Oracle Secure Backup **administrative domain** requires that you first configure the **administrative server** and then configure each **media server** and **client**.

Configuring the Administrative Server

If you install Oracle Secure Backup on a host and specify this host as the **administrative server**, then this server is the **Certification Authority (CA)** for the Oracle Secure Backup **administrative domain**. Oracle Secure Backup configures the host as the CA automatically as part of the standard installation. You are not required to take additional steps to provide a signing **certificate** for this server.

Oracle Secure Backup automatically creates the following items:

- A host object corresponding to the administrative server in the object repository on the administrative server.
- A **wallet** to contain the administrative server's certificates. The wallet resides in the directory tree of the **Oracle Secure Backup home**. Oracle Secure Backup uses the host ID as the wallet password.
- A request for a signing certificate in the wallet.
- A signed certificate in response to the request and stores the certificate in the wallet.
- A request for an **identity certificate** in the wallet.
- A signed certificate in response to the request and stores it in the wallet.
- An **obfuscated wallet** in the local wallet directory.

The administrative server now has the signing certificate, which it must have to sign the identity certificates for other hosts, and its identity certificate, which it must have

to establish authenticated **Secure Sockets Layer (SSL)** connections with other hosts in the domain.

Configuring Media Servers and Clients

Oracle Secure Backup creates security credentials for a host when you use the Oracle Secure Backup **Web tool** or run the `mkhost` command in **obtool** to configure the host. The procedure differs depending on whether you add hosts in automated or **manual certificate provisioning mode**.

> **See Also:** "Determining the Distribution Method of Host Identity Certificates" on page 6-7

Automated Certificate Provisioning Mode

If you create the hosts in **automated certificate provisioning mode**, then you are not required to perform additional steps. Oracle Secure Backup creates the **wallet**, keys, and certificates for the host automatically as part of the normal host configuration.

Manual Certificate Provisioning Mode

You must use the obcm utility when you add hosts in the domain in manual rather than **automated certificate provisioning mode**. In this case, the certificate authority does not issue a signed certificate to a host over the network, so you must export the signed certificate from the **administrative server**, manually transfer the certificate to the newly configured host, and then import the certificate into the host's **wallet**.

Both an **identity certificate** and a wallet exist as files on the operating system. The operating system user running obcm must have write permissions in the wallet directory. By default, the wallet used by Oracle Secure Backup is located in the following locations:

- `/usr/etc/ob/wallet` (UNIX and Linux)
- `C:\Program Files\Oracle\Backup\db\wallet` (Windows)

The obcm utility always accesses the wallet in the preceding locations. You cannot override the default location.

If you choose to add hosts in **manual certificate provisioning mode**, then you must perform the following steps for each host:

1. Log on to the administrative server.
2. Assuming that your PATH variable is set correctly, enter `obcm` at the operating system command line to start the obcm utility. The operating system user running obcm must have write permissions in the wallet directory.
3. Enter the following command, where *`hostname`* is the name of the host requesting the certificate and *`certificate_file`* is the file name of the exported request:

   ```
   export --certificate --file certificate_file --host hostname
   ```

   For example, the following command exports the signed certificate for host `brhost2` to file `/tmp/brhost2_cert.f`:

   ```
   export --certificate --file /tmp/brhost2_cert.f --host brhost2
   ```

4. Copy the signed identity certificate to some type of physical media and physically transfer the media to the host.
5. Log on to the host whose wallet contains the certificate.

6. Assuming that your PATH variable is set correctly, enter `obcm` at the operating system command line to start the obcm utility. The operating system user running obcm must have write permissions in the wallet directory.
7. Copy the signed identity certificate to a temporary location on the file system.
8. Enter the following command at the obcm prompt, where *`signed_certificate_file`* is the file name of the certificate:

   ```
   import --file signed_certificate_file
   ```

   Because only one Oracle Secure Backup wallet exists on the host, you are not required to specify the `--host` option. For example, the following example imports the certificate from `/tmp/brhost2_cert.f`:

   ```
   import --file /tmp/brhost2_cert.f
   ```

   The obcm utility issues an error message if the certificate being imported does not correspond to the certificate request in the wallet.
9. Remove the certificate file from its temporary location on the operating system. For example:

   ```
   rm /tmp/brhost2_cert.f
   ```

The obcm utility checks that the **public key** associated with the certificate for the host corresponds to the **private key** stored in the wallet with the certificate request. If the keys match, then the host is a member of the domain. If the keys do not match, then an attacker probably attempted to pass off their own host as the host during processing of the `mkhost` command. You can run the `mkhost` command again after the rogue host has been eliminated from the network.

Setting the Size for Public and Private Keys

As a general rule, the larger the sizes of the **public key** and the **private key**, the more secure they are. On the other hand, the smaller the key, the better the performance. The default key size for all hosts in the domain is 1024 bits. If you accept this default, then you are not required to perform any additional configuration.

Oracle Secure Backup enables you to set the key to any of the following bit values, which are listed in descending order of security:

- 4096
- 3072
- 2048
- 1024
- 768
- 512

This section contains these topics:

- Setting the Key Size in obparameters
- Setting the Key Size in the certkeysize Security Policy
- Setting the Key Size in mkhost

Setting the Key Size in obparameters

The `obparameters` file specifies the default key size in the security policy, which if used is set up during the installation process. The key size for all hosts in the domain defaults to this value.

You can set the key size in the `obparameters` file when you install Oracle Secure Backup on the **administrative server**. When you install Oracle Secure Backup interactively, the install script gives you an opportunity to modify the `obparameters` file.

To set the key size in `obparameters` when installing interactively:

1. Before running the install script on the administrative server, or when the install script prompts you to modify `obparameters`, open the file in a text editor.
2. Search for the following string: `certificate key size`. Set the key size to the desired default value. The following example sets the default key size to 2048 bits:

   ```
   identity certificate key size: 2048
   ```

3. Save and close the file after making any other changes to `obparameters`.
4. Proceed with the installation.

Oracle Secure Backup uses the key size in `obparameters` to set the initial value for the `certkeysize` security policy. This security policy specifies the default security key size for hosts in the domain. You can change or override this default when configuring an individual host.

Note: There is no equivalent procedure for Windows. Windows users are restricted to the default value.

See Also: Appendix B, "Oracle Secure Backup obparameters Installation Parameters"

Setting the Key Size in the certkeysize Security Policy

You can change the default key size in the security policy at any time. Any hosts configured after the change default to the changed key size.

You can set the key size in the `certkeysize` security policy through **obtool** or the Oracle Secure Backup **Web tool**. Oracle Secure Backup uses the modified key size the next time you configure a host. You can change the `certkeysize` value at any time, but the change only applies to the next `mkhost` command.

To set the `certkeysize` security policy:

1. Log in to `obtool` as a user with the `modify administrative domain's configuration` right.
2. Set the `certkeysize` policy to the desired default value. The following example shows how to use `obtool` to set the key size to 3072 bits:

   ```
   ob> cdp security
   ob> setp certkeysize 3072
   ```

See Also: *Oracle Secure Backup Administrator's Guide* to learn how to set a policy

Setting the Key Size in mkhost

You can override the default key size for any individual host. Different hosts in the domain can have different key sizes.

You can set the key size when you use the `mkhost` command or Oracle Secure Backup **Web tool** to configure a host. If you specify the `--certkeysize` option on the `mkhost` command, then the specified value overrides the default certificate key size set in the security policy. The key size applies only to the newly configured host and does not affect the key size of any other current or future hosts.

Because larger key sizes require more computation time to generate the key pair than smaller key sizes, the key size setting can affect the processing time of the `mkhost` command. While the `mkhost` command is running, **obtool** might display a status message every 5 seconds. `obtool` displays a command prompt when the process has completed.

To set the key size in the `mkhost` command:

1. Log in to `obtool` as a user with the `modify administrative domain's configuration` right.
2. Issue the `mkhost` command to set the key size for a host. The following example sets the key size to 4096 bits when configuring **client** stadf56. This setting applies only to host stadf56.

```
ob> mkhost --inservice --role client --certkeysize 4096 stadf56
Info: waiting for host to update certification status...
Info: waiting for host to update certification status...
Info: waiting for host to update certification status...
Info: waiting for host to update certification status...
ob> lshost stadf56
stadf56          client                           (via OB)   in service
```

> **See Also:** *Oracle Secure Backup Reference* to learn how to use the `mkhost` command

Enabling and Disabling SSL for Host Authentication and Communication

By default Oracle Secure Backup uses authenticated and encrypted **Secure Sockets Layer (SSL)** connections for all control message traffic among hosts.

You can disable SSL encryption by setting the `securecomms` security policy to `off`. Disabling SSL might improve performance, but be aware of the inherent security risks in this action.

> **See Also:** "Host Authentication and Communication" on page 6-9

To set the `securecomms` security policy:

1. Log in to **obtool** as a user with the `modify administrative domain's configuration` right.
2. Use the `setp` command to switch the `securecomms` policy to `off`, as shown in the following example:

```
ob> cdp security
ob> setp securecomms off
```

See Also: *Oracle Secure Backup Administrator's Guide* to learn how to set a policy

Managing Certificates with obcm

This section explains how to use the obcm utility. You can use this utility to import certificates, export certificates, and export certificate requests.

You must use obcm when you add hosts in the domain in manual rather than **automated certificate provisioning mode**. In this case, the **Certification Authority (CA)** does not issue a signed certificate to a host over the network, so you must export the signed certificate from the **administrative server**, manually transfer the certificate to the newly configured host, and then import the certificate into the host's **wallet**.

Both an **identity certificate** and a wallet exist as files on the operating system. The operating system user running obcm must have write permissions in the wallet directory. By default, the wallet used by Oracle Secure Backup is located in the following locations:

- `/usr/etc/ob/wallet` (UNIX and Linux)
- `C:\Program Files\Oracle\Backup\db\wallet` (Windows)

The obcm utility always accesses the wallet in the preceding locations. You cannot override the default location.

Exporting Signed Certificates

You can use obcm on the **administrative server** to export a signed **certificate** for a newly configured host.

To export a signed **identity certificate**:

1. Log on to the administrative server.
2. Assuming that your PATH variable is set correctly, enter `obcm` at the operating system command line to start the obcm utility. The operating system user running obcm must have write permissions in the **wallet** directory.
3. Enter the following command, where *`hostname`* is the name of the host requesting the certificate and *`certificate_file`* is the file name of the exported request:

   ```
   export --certificate --file certificate_file --host hostname
   ```

   For example, the following command exports the signed certificate for host `brhost2` to file `/tmp/brhost2_cert.f`:

   ```
   export --certificate --file /tmp/brhost2_cert.f --host brhost2
   ```

Importing Signed Certificates

You can use obcm on the host to import a signed **certificate** into the host's **wallet**.

To import a signed **identity certificate** into the wallet of a host:

1. Log on to the host whose wallet contains the certificate.
2. Assuming that your PATH variable is set correctly, enter `obcm` at the operating system command line to start the obcm utility. The operating system user running obcm must have write permissions in the wallet directory.
3. Copy the signed identity certificate to a temporary location on the file system.

4. Enter the following command at the obcm prompt, where *signed_certificate_file* is the file name of the certificate:

   ```
   import --file signed_certificate_file
   ```

   Because only one Oracle Secure Backup wallet exists on the host, you are not required to specify the `--host` option. For example, the following example imports the certificate from `/tmp/brhost2_cert.f`:

   ```
   import --file /tmp/brhost2_cert.f
   ```

   The obcm utility issues an error message if the certificate being imported does not correspond to the certificate request in the wallet.

5. Remove the certificate file from its temporary location on the operating system. For example:

   ```
   rm /tmp/brhost2_cert.f
   ```

A

Oracle Secure Backup Directories and Files

This appendix explains the structure and contents of the Oracle Secure Backup directories.

This appendix contains these sections:

- Oracle Secure Backup Home Directory
- Administrative Server Directories and Files
- Media Server Directories and Files
- Client Host Directories and Files

> **Note:** Some of the directories and files listed in this appendix are not created until after a backup has been performed by Oracle Secure Backup.

Oracle Secure Backup Home Directory

When you installed Oracle Secure Backup, you specified an **Oracle Secure Backup home** directory for the installation. Oracle recommends the following locations for the Oracle Secure Backup home:

- `C:\Program Files\Oracle\Backup` on Windows
- `/usr/local/oracle/backup` on Linux and UNIX

The Oracle Secure Backup home directory is created on every host where you install Oracle Secure Backup, although the contents of the directory vary depending on the **roles** you assigned to the host.

Each host on which Oracle Secure Backup is installed contains a configuration file that records details of the configuration of Oracle Secure Backup on the host. On Windows, the configuration file is called obconfig.txt in the db subdirectory of the Oracle Secure Backup home. On Linux and UNIX, the file is called obconfig and is located in the `/etc` directory.

Administrative Server Directories and Files

An **administrative server** contains a set of executables and data files for each installed operating system, which are described in the following tables:

- Architecture-Independent Directories and Files for an Administrative Server
- Windows Directories for an Administrative Server

- Linux and UNIX Directories and Files for an Administrative Server

Table A–1 Architecture-Independent Directories and Files for an Administrative Server

Directory or File	Description
admin/	Administrative domain databases
admin/config/	Configuration databases
admin/config/class/	User class data
admin/config/dataset/	Datasets
admin/config/default/	Defaults and policies data
admin/config/device/	Device data
admin/config/duplication/	Duplication data
admin/config/family/	Media family data
admin/config/host/	Host data
admin/config/location/	Vaulting location data
admin/config/rotation/	Volume rotation data
admin/config/schedule/	Backup schedules
admin/config/summary/	Summary data
admin/config/user/	User data
admin/encryption/	Encryption data
admin/encryption/keys/	Keys used in encryption
admin/encryption/wallet/	Wallet used in encryption
admin/history/	History data generated by Oracle Secure Backup
admin/history/edcf/	Network Data Management Protocol (NDMP) environment data container files
admin/history/host/	Host-specific history data
admin/history/host/*host_name*/	Backup catalog for *host_name*
admin/log/	Generated log files
admin/log/device/	Log files for devices
admin/log/device/*device_name*/	Log files for *device_name*
admin/log/index/	Backup catalog manager logs
admin/log/scheduler/	Scheduler-generated logs
admin/log/scheduler/summary/	Log files for email summary reports
admin/log/security/	Security-related logfiles
admin/state/	Dynamic state data
admin/state/device/	Device state
admin/state/device/*device_name*/	State for *device_name*
admin/state/family/	Media family state
admin/state/family/*media_family_name*	State for *media_family_name*
admin/state/general/	Miscellaneous state
admin/state/host/	Host state

Table A–1 (Cont.) Architecture-Independent Directories and Files for an Administrative Server

Directory or File	Description
admin/state/host/*host_name*/	State for *host_name*
admin/state/scheduler/	Scheduler state
admin/state/scheduler/job/	Job state
apache/	Apache Web server files
apache/conf/	Apache server configuration files
apache/conf/ssl.crl/	Apache server certificate revocation list
apache/conf/ssl.crt/	Apache server certificate
apache/conf/ssl.csr/	Apache server certificate signing request
apache/conf/ssl.key/	Apache server SSL key
apache/conf/ssl.prm/	Apache server public DSA parameter files
apache/htdocs/	Apache server HTML document root
apache/htdocs/css/	Apache server custom style sheets
apache/htdocs/include/	Apache server PHP files
apache/htdocs/include/policies/	Apache server PHP files
apache/htdocs/js/	Apache server Java script files
apache/htdocs/php/	Apache server PHP files
apache/images/	Apache server Web image files
apache/logs/	Apache server log files
bin/	Executables or links to executables: ■ In an installation on a Windows operating system, this directory contains the executables for the Windows operating system. ■ In an installation on a Linux or UNIX operating system, this directory contains links to the executables for the operating system.
device/	Device tables
help/	Oracle Secure Backup help files
samples/	Sample tools for scripting with Oracle Secure Backup

Table A–2 Windows Directories for an Administrative Server

Directory	Description
db\xcr\	Transcripts for jobs that ran on this host
db\.hostid	Identifying information for this host
db\wallet	Security credentials for this host
temp\	Log file for observiced and temporary files

Table A–3 Linux and UNIX Directories and Files for an Administrative Server

Directory or File	Description
.bin.*operating_system*/	Executables for *operating_system*, where *operating_system* is a derivative of the operating system name. For example, the directory for Sun Solaris is .bin.solaris.
.drv.*operating_system*/	Device drivers for *operating_system*
etc/	Architecture-independent executables for daemons and maintenance tools
.etc.*operating_system*/	Daemons and utility programs for *operating_ system*
install/	Installation programs
lib/	Architecture-independent shared library for the system backup to tape (SBT) interface
.lib.*operating_system*/	Shared library for the SBT interface for *operating_system*, where *operating_system* is a derivative of the operating system name. For example, the directory for Sun Solaris is .lib.solaris.
man/	Man pages for Oracle Secure Backup components
man/man1	Man pages for Oracle Secure Backup executables
man/man8	Man pages for daemons and maintenance tools
tools.*operating_system*/	Maintenance tools
/usr/etc/ob/.hostid	Identifying information for this host
/usr/etc/ob/wallet	Security credentials for this host
/usr/etc/ob/xcr/	Transcripts for jobs that ran on this host
/usr/tmp/	Log files for observiced files, obndmpd files, and temporary files
.wrapper	Shell program that selects an executable from a .bin.* or .etc.* directory, based on the computer architecture of the host executing the command. Symbolic links and the architecture-independent .wrapper shell program enable hosts to contain executables for multiple computer architectures.

Media Server Directories and Files

Every **media server** contains a subset of the directories and files found on an **administrative server**. The only files included are those pertinent to the computer architecture of the server and its function as a media server and **client**. They are described in the following tables:

- Architecture-Independent Directories for a Media Server
- Windows Directories for a Media Server
- Linux and UNIX Directories and Files for a Media Server

Table A–4 Architecture-Independent Directories for a Media Server

Directory	Description
bin/	Executables or links to executables: ■ In an installation on a Windows operating system, this directory contains the executables for the Windows operating system. ■ In an installation on a Linux or UNIX operating system, this directory contains links to the executables for the operating system.
device/	Device tables

Table A–5 Windows Directories for a Media Server

Directory	Description
drv\	Device driver
help\	Oracle Secure Backup help files
temp\	Log file for observiced and temporary files
db\.hostid	Identifying information for this host
db\wallet	Security credentials for this host

Table A–6 Linux and UNIX Directories and Files for a Media Server

Directory or File	Description
.bin.*operating_system*/	Executables for *operating_system*, where *operating_system* is a derivative of the operating system name. For example, the directory for Sun Solaris is .bin.solaris.
.drv.*operating_system*/	Device drivers for *operating_system*
etc/	Architecture-independent executables for daemons and maintenance tools
.etc.*operating_system*/	Daemons and utility programs for *operating_ system*
man/	Man pages for Oracle Secure Backup components
/usr/etc/ob/.hostid	Identifying information for this host
/usr/etc/ob/xcr/	Transcripts for jobs that ran on this host
/usr/tmp/	Log files for observiced files, obndmpd files, and temporary files
.wrapper	Shell program that selects an executable from a .bin.* or .etc.* directory, based on the computer architecture of the host executing the command. Symbolic links and the architecture-independent .wrapper shell program enable hosts to contain executables for multiple computer architectures.

Client Host Directories and Files

Every computer that acts only as a **client** host contains the minimum set of directories and files needed for Oracle Secure Backup operations. They are described in the following tables:

- Architecture-Independent Directory for a Client Host
- Windows Directories and Files for a Client Host
- Linux and UNIX Directories and Files for a Client Host

Table A–7 Architecture-Independent Directory for a Client Host

Directory	Description
bin/	Executables or links to executables ■ In an installation on a Windows operating system, this directory contains the executables for the Windows operating system. ■ In an installation on a Linux or UNIX operating system, this directory contains links to the executables for the operating system.

Table A–8 Windows Directories and Files for a Client Host

Directory	Description
db\.hostid	Identifying information for this host
db\wallet	Security credentials for this host.
temp\	Log file for observiced and temporary files
help\	Oracle Secure Backup help files

Table A–9 Linux and UNIX Directories and Files for a Client Host

Directory or File	Description
.bin.*operating_system*/	Executables for *operating_system*, where *operating_system* is a derivative of the operating system name. For example, the directory for Sun Solaris is .bin.solaris.
etc/	Architecture-independent executables for daemons and maintenance tools
.etc.*operating_system*/	Daemons and utility programs for *operating_ system*
man/	Man pages for Oracle Secure Backup components
/usr/etc/ob/.hostid	Identifying information for this host
/usr/etc/ob/xcr/	Transcripts for jobs that ran on this host
/usr/tmp/	Log files for observiced files, obndmpd files, and temporary files
.wrapper	Shell program that selects an executable from a .bin.* or .etc.* directory, based on the computer architecture of the host executing the command. Symbolic links and the architecture-independent .wrapper shell program enable hosts to contain executables for multiple computer architectures.

B

Oracle Secure Backup obparameters Installation Parameters

This appendix describes the installation parameters for Oracle Secure Backup on Linux or UNIX. You can set these parameters in the `obparameters` file, which is a plain text file located in the install subdirectory of the Linux or UNIX **Oracle Secure Backup home**.

> **Note:** The `obparameters` file is not used in Windows installations.

This appendix contains these sections:

customized obparameters

If you customize any of the parameters in the `obparameters` file, then set the `customized obparameters` parameter to `yes`.

Table B–1 customized obparameters: Values

Value	Meaning
`no` (default)	Specifies that installation parameters in the `obparameters` file have not been changed. The value of `no` is set by default.
`yes`	Specifies that installation parameters in the `obparameters` file have been changed.

start daemons at boot

The installation tools can update the control file of each host to automatically start Oracle Secure Backup each time you start the system.

Table B–2 start daemons at boot: Values

Value	Meaning
`no`	Specifies that the Oracle Secure Backup daemons do not start automatically at start time.
`yes` (default)	Specifies that the Oracle Secure Backup daemons start automatically at start time.

identity certificate key size

This option configures the key size in bits, and thus the level of security, associated with every host **identity certificate** issued by the administrative **service daemon**.

The default is `1024`.

> **Note:** Certificate key sizes smaller than 1024 are not considered secure. Certificate key sizes of 3072 or more are considered very secure.

Table B–3 identity certificate key size: Values

Value	Meaning
`512`	Specifies a 512-bit long certificate key size.
`768`	Specifies a 768-bit long certificate key size.
`1024` (default)	Specifies a 1024-bit key length. This is the minimum required value for adequate security.
`2048`	Specifies a 2048-bit key length. This value offers adequate security.
`3072`	Specifies a 3072-bit key length. This value offers a very high level of security.
`4096`	Specifies a 4096-bit key length. This value offers a very high level of security.

create preauthorized oracle user

This parameter controls whether the Oracle Secure Backup installation process creates an **Oracle Secure Backup user** named `oracle` which has been preauthorized to perform database backup and restore operations.

Table B–4 create preauthorized oracle user: Values

Value	Meaning
`yes`	An Oracle Secure Backup user is created during installation. The parameters `default UNIX user` and `default UNIX group` specify the user and group parameters with which the Oracle Secure Backup user is created.
`no` (default)	No Oracle user is created.

default UNIX user

After the Oracle Secure Backup installation is successfully completed and the **administrative domain** has been initialized, you can create a default **Oracle Secure Backup user** named `oracle` if requested. By setting this parameter, you specify the Linux or UNIX operating system user to which the Oracle Secure Backup user named `oracle` is mapped. You can also perform this task through the Oracle Secure Backup **Web tool**.

See Also: "create preauthorized oracle user" on page B-2

Table B–5 default UNIX user: Values

Value	Meaning
`UNIX_user`	Specifies the Linux or UNIX operating system user name defined in `/etc/password` to which the Oracle Secure Backup user named `oracle` is mapped. By default, the Linux/UNIX user is named `oracle`.

default UNIX group

After the installation is successfully completed and the **administrative domain** has been initialized, a default group is created on Linux or UNIX if requested. The user specified by the `default UNIX user` parameter is a member of this group.

See Also: "create preauthorized oracle user" on page B-2

Table B–6 default UNIX group: Values

Value	Meaning
`UNIX_group`	Specifies a Linux or UNIX group defined in `/etc/group`. By default, the Linux/UNIX group is `dba`.

linux ob dir and solaris64 ob dir

To keep the installation and administration of Oracle Secure Backup as straightforward as possible, Oracle provides a mechanism for you to identify the name of the **Oracle Secure Backup home** directory for each platform in your network. This directory must be private to each platform and not shared through **Network File System (NFS)** or a similar remote file system.

When the installation programs install Oracle Secure Backup software, they choose these home directories for the installation or verify that these are the directories you have used. These defaults might be changed based on the availability of disk space on your computer.

Table B–7 os-name ob dir: Parameters and Values

Parameter	Meaning
`linux ob dir`	Specifies Oracle Secure Backup home location for Linux hosts. The default is `/usr/local/oracle/backup`.
`solaris64 ob dir`	Specifies Oracle Secure Backup home location for Solaris 64-bit hosts. The default is `/usr/local/oracle/backup`.

linux db dir and solaris64 db dir

Each platform has a discrete directory in which Oracle Secure Backup retains host-specific information. This directory must be private to each platform and not shared through **Network File System (NFS)** or a similar remote file system.

Table B–8 os-name db dir: Parameters and Values

Parameter	Meaning
`linux db dir`	Specifies the directory where host-specific information is retained for Linux hosts. The default directory is `/usr/etc/ob`.
`solaris64 db dir`	Specifies the directory where host-specific information is retained for Solaris 64-bit hosts. The default directory is `/usr/etc/ob`.

linux temp dir and solaris64 temp dir

Oracle Secure Backup typically uses the `/usr/tmp` directory on each host for storage of transient files. Oracle Secure Backup requires that the temporary directory be able to contain lockable files and that it be accessible during the beginning of the restart process. The directory must be on the local disk. You can specify a different directory for each platform by modifying any of these `<os-name> temp dir` parameters.

Table B–9 os-name temp dir: Parameters and Values

Parameter	Meaning
`linux temp dir`	Specifies the directory where transient files are stored for Linux hosts. The default directory is `/usr/tmp`.
`solaris64 temp dir`	Specifies the directory where transient files are stored for Solaris 64-bit hosts. The default directory is `/usr/tmp`.

linux links and solaris64 links

During installation, symbolic links are created, typically in `/usr/bin` and `/etc`, so that an **Oracle Secure Backup user** is not required to change search paths. You can modify this behavior as follows:

- Comment out or delete these parameters if you do not want the installation programs to create any links.
- Change the value of these parameters if you want the installation programs to create links in another directory for a specific platform.

These parameters are particular to each supported platform. On some systems, it might be more appropriate to place links in `/bin` instead of `/usr/bin` or in `/usr/etc` instead of `/etc`.

This parameter must be followed by three values, in the order shown:

1. The name of the directory in which to create the bin link.
2. The name of the directory in which to create the etc link.
3. The name of the directory in which to create the lib link.

Note: Oracle recommends using the defaults provided for this parameter.

Table B–10 os-name links: Parameters and Values

Parameter	Meaning
`linux links`	Specifies the directories where symbolic links are created for Linux hosts. The default directory list is `/usr/bin/etc/lib`.
`solaris64 links`	Specifies the directories where symbolic links are created for Solaris 64-bit hosts. The default directory list is `/usr/bin/etc/lib`.

Note: If the `obparameters` file specifies a lib directory for the operating system type of the current installation, then `installob` creates a libobk.so symbolic link in that directory. That symbolic link points to the actual libobk.so file in a platform-specific lib directory in the **Oracle Secure Backup home** (such as.lib.linux32).

ask about ob dir

Specifies whether the installation notifies you when you are about to install Oracle Secure Backup into a directory other than the default **Oracle Secure Backup home**.

Table B–11 ask about ob dir: Values

Value	Meaning
`yes`	Enables notification when you select a directory other than the default Oracle Secure Backup home.
`no` (default)	Suppresses notification when you select a directory other than the default Oracle Secure Backup home.

default protection

Specifies directory and file protection information that is in effect when the Oracle Secure Backup installation is complete.

Caution: The file protection information is provided for reference only. Oracle strongly recommends using the defaults provided because changing them can prevent the product from functioning.

Values

Each line in the `default protection` section of the `obparameters` file indicates the file owner, group number and permissions for the file or files specified by name, or by wildcard pattern. The default values are as follows:

```
default protection:
```

```
root.0          755 ./.wrapper
root.0          644 ./device/*
root.0          755 ./install/*
root.0          644 ./help/*
root.0          755 ./man/*
root.0          644 ./man/man1/*
root.0          644 ./man/man8/*
root.0          644 ./samples/*
root.0          755 ./samples/autoobtar
root.0          755 ./samples/bdf2ds
root.0          755 ./samples/*.sh
root.0          700 ./admin
root.0          700 ./admin/*
root.0          700 ./admin/config/*
root.0          755 ./.bin.*/*
root.0         4755 ./.bin.*/obtar
root.0         4755 ./.bin.*/obt
root.0         4755 ./.bin.*/obtool
root.0          755 ./.etc.*/*
root.0         4755 ./.etc.*/obixd
root.0         4755 ./.etc.*/observiced
root.0         4755 ./.etc.*/obscheduled
root.0         4755 ./.etc.*/obrobotd
root.0          755 ./.etc.*/
root.0         4755 ./.etc.*/doswitch
root.0          644 ./.drv.*/*
root.0          755 ./.lib.*/*
root.0          755 ./*
root.0          755 /usr/etc/ob
root.0          644 /usr/etc/ob/.hostid
root.0          755 /usr/etc/ob/xcr
root.0          644 /etc/obconfig
```

run obopenssl

Specifies whether the installation prompts you to create the certificates for the **Apache Web server**.

> **Note:** Oracle recommends using the default provided to ensure proper initialization of your Oracle Secure Backup **Web tool**.

Table B–12 run obopenssl: Values

Value	Meaning
`yes` (default)	Create the certificate.
`no`	Do not create the certificate.

C

Determining Linux SCSI Parameters

For the Linux platforms, if you do not know the **SCSI** parameters of a **tape device**, then you must determine them before you begin installation. This appendix describes procedures for determining SCSI device parameters on Linux.

Determining SCSI Device Parameters on Linux

To obtain tape device information on Linux, use the `cat` command to view the contents of `/proc/scsi/scsi`. For example:

```
# cat /proc/scsi/scsi
```

> **See Also:** "Identifying and Configuring Linux Attach Points" on page 2-18 for information about configuring attach points for Linux

Example C–1 shows sample output for a host called `storabck05` with two attached tape devices.

Example C–1 Sample /proc/scsi/scsi Contents

```
Attached devices:
Host: scsi0 Channel: 00 Id: 02 Lun: 00
  Vendor: IBM      Model: ULTRIUM-TD2      Rev: 4772
  Type:   Sequential-Access                ANSI SCSI revision: 03
Host: scsi0 Channel: 00 Id: 04 Lun: 00
  Vendor: ADIC     Model: Scalar 24        Rev: 237A
  Type:   Medium Changer                   ANSI SCSI revision: 02
```

A device of type `Sequential-Access`, such as the first tape device in the list, is a **tape drive**. A device of type `Medium Changer`, such as the second tape device, is a **tape library**.

For each tape device, the information needed is found in the line that reads:

```
Host: scsi0 Channel: 00 Id: 02 Lun: 00
```

The output can be interpreted as follows:

- The host bus adapter number is the numeric part of the value `scsi`*n*. For example, for both tape devices in this output the host bus adapter number is `0`.
- The value for `Channel` is the SCSI bus address. For example, in this output the SCSI bus address is `0`.
- The value for `Id` is the target ID. For example, in this output the ID of the tape drive is `2`, and the ID of the tape library is `4`.

- The value for `Lun` is the **SCSI LUN**. For example, in this output the SCSI LUN of both tape devices is `0`.

By convention, the tape library and tape drive can each be assigned `0` as the **Oracle Secure Backup logical unit number**.

Based on the output shown in Example C–1, Table C–1 summarizes the tape device information for storabck05.

Table C–1 Device Summary

Device	Host Bus Adapter	SCSI bus address	Target ID	SCSI LUN
Library	0	0	2	0
Tape drive	0	0	4	0

D

Oracle Secure Backup and ACSLS

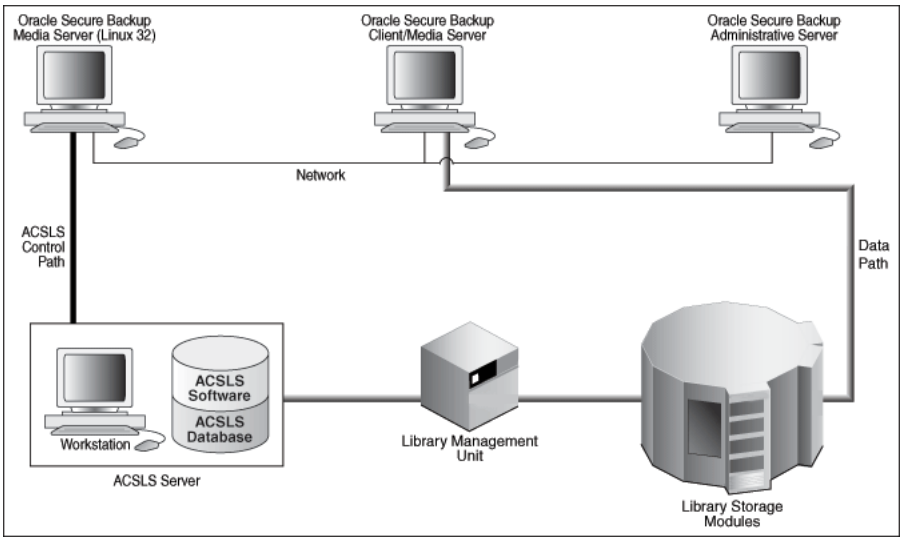
This appendix describes Oracle Secure Backup support for StorageTek Automated Cartridge System Library Software (ACSLS). ACSLS is a package of server software that controls one or more Automated Cartridge Systems **tape library**.

This appendix contains these sections:

About ACSLS

Figure D–1 shows how ACSLS fits into a configuration of client systems, Library Storage Modules (LSMs), and a single Library Management Unit (LMU). The LSM is hardware that has cartridge slots, a robotic arm, pass through ports, cartridge access ports, and the **tape drive**. The LMU is the hardware interface between the ACSLS and the LSM.

Figure D–1 Library with ACSLS Server



ACSLS offers the following advantages:

- Handles multiple libraries and multiple clients
- Manages tape drive loading and unloading
- Manages tape **volume** importing and exporting
- Handles mixed media types
- Optionally imposes access controls based on user ID, command, and **volume ID**
- Supports multiple pools of scratch tapes
- Generates inventory and configuration reports
- Manages cleaning cartridges and cleaning operations

ACSLS and Oracle Secure Backup

An ACSLS **volume** is called a cartridge. Cartridges are loaded and unloaded through cartridge access points. Oracle Secure Backup **obtool** device commands `mkdev`, `chdev`, `lsdev`, and `rmdev` have been modified to manage these cartridge access points.

Note: ACSLS can be controlled using `obtool` only. Neither the Oracle Secure Backup **Web tool** nor Oracle Enterprise Manager is supported for ACSLS.

See Also:

- "Modified Oracle Secure Backup Commands" on page D-4
- *Oracle Secure Backup Reference* for more information on `obtool` device commands

ACSLS references all of its volumes by their external **barcode** labels, which are required for all ACS volumes. Oracle Secure Backup continues to allow the **operator** to access these ACS volumes by **storage element**, **volume label**, and barcode label.

> **Note:** ACSLS supports *virtual tapes* that do not have a physical barcode attached to them. Oracle Secure Backup does not support virtual tapes within an ACS system. Oracle Secure Backup requires that all cartridges within an ACS system have properly affixed and readable barcodes.

The concept of a scratch pool in ACSLS is simply a blank tape. Once a tape has been mounted in a **tape drive**, its scratch pool identity is removed, and it acquires a permanent **media family**, identical in functionality to the pre-labeling volumes. Oracle Secure Backup supports scratch pools through an extension to the media family and retains this concept through the existing media family functionality. In addition, when a volume is force unlabeled it is moved back into the scratch pool that is assigned to the media family.

ACSLS has optional access control mechanisms on commands and volumes. This optional access control user ID can be defined as part of the `mkdev` or `chdev` commands.

Because an ACSLS system is meant to be shared by multiple clients, tape drive cleaning is managed and maintained by ACSLS.

Communicating with ACSLS

Oracle Secure Backup uses the `obrobotd` daemon when talking to a non-ACSLS **tape library**. When talking with an ACSLS tape library, Oracle Secure Backup uses two **daemons** named obacslibd and obacsssid. The obacslibd daemon spawns obacsssid, which is responsible for communications with the ACSLS server.

Drive Association

When you install a **tape drive** other than an ACS tape drive, Oracle Secure Backup requires that you attach the tape drive to a **media server**, install an appropriate operating system driver for the tape drive, create a device within Oracle Secure Backup, and map the operating system device to the Oracle Secure Backup device. The same steps are required for ACSLS. But you must also further define the ACSLS mapping of the tape drive through the `mkdev` or `chdev` command. The additional information required is the `acs`, `lsm`, `panel`, and `drive`.

Volume Loading and Unloading

Drive identification for mounts and dismounts is by **tape drive** name.

ACSLS always identifies a **volume** by its **barcode**. Because Oracle Secure Backup associates this barcode with a **volume ID**, you can supply either one. If a mapping is not possible, then the request is rejected with appropriate logging.

Imports and Exports

The `exportvol` command has been modified to conform to ACSLS usage. Individual ACS cartridge access port (CAP) slots are not addressable, although an entire CAP can be selected based on CAP name.

Once the request is made to eject the tape, the request does not return until the CAP has been opened, the cartridge loader emptied, and the cartridge loader reinserted in

that emptied state. Because there is only one obacslibd daemon controlling each ACS **tape library**, no other tape library operations are permitted until the CAP is cleared. You can control how long an outstanding request waits for the CAP to be cleared with the `maxacsejectwaittime` policy.

Oracle Secure Backup does not support the `importvol` command for ACSLS tape libraries. You can use the ACSLS cmd_proc utility to enter a **volume** into the tape library.

Access Controls

ACSLS optionally allows fine-grained access control over the commands that a user can issue and the volumes that can be accessed. Setting up the ACSLS access controls is done at the ACSLS console. Oracle Secure Backup does not support setting, modifying, or displaying the ACSLS access controls.

If ACSLS access control is enabled, then a user must have the correct `acsls_access_id` to access the ACS device. Oracle Secure Backup maps this `acsls_access_id`, which is defined on the **obtool** `mkdev` or `chdev` commands, to the Oracle Secure Backup device object.

Scratch Pool Management

ACSLS enables you to define one or more scratch pools to which a blank or recycled **volume** can be assigned. Subsequent scratch mount requests are then restricted to volumes in the pool or pools specified with the request. Oracle Secure Backup offers equivalent functionality with an optional scratch pool ID for **media family** objects.

When a volume is pulled from the scratch pool, Oracle Secure Backup automatically labels the volume with a permanent media family when its volume header is written. You are not required to label volumes with the `labelvol` command beforehand. This ensure that separation of tapes within the tape libraries is persistent.

When an `unlabelvol` operation is performed, the tape is put back into the scratch pool that is defined within the current definition of the media family.

Oracle Secure Backup does not support creating scratch pools, entering cartridges into a scratch pool, or removing cartridges from a scratch pool. These operations must be performed at the ACSLS console.

Modified Oracle Secure Backup Commands

The following Oracle Secure Backup commands are modified for ACSLS tape libraries:

- mkdev
- chdev
- lsdev
- exportvol
- mkmf
- chmf

> **See Also:** *Oracle Secure Backup Reference* for syntax and semantics for device, library, and **media family obtool** commands

Unsupported Oracle Secure Backup Commands

The following Oracle Secure Backup commands are not supported for ACSLS tape libraries:

- `importvol`
- `extractvol`
- `insertvol`
- `clean`
- `opendoor`
- `closedoor`

Installation and Configuration

The Oracle Secure Backup **media server** attached to the ACSLS server must be either a Linux x86-64 bit media server or a Linux 32-bit media server.

Oracle Secure Backup installation assumes that the ACSLS hardware and software has been correctly installed and configured. Oracle Secure Backup installation procedures do not attempt to create or modify any ACSLS configuration files.

Oracle Secure Backup handles ACS tape devices no differently from other devices. The Oracle Secure Backup device driver (if any) is installed, and special device files are created. The data path is controlled solely by Oracle Secure Backup. ACSLS is not involved.

creating Oracle Secure Backup objects for ACSLS devices is performed with the `mkdev` command in **obtool** with the following modifications:

- For ACSLS tape libraries, the usual host:devname attach point is replaced with information identifying the `acs` of the tape library and the host name and port where the associated ACS software is listening. A **barcode** reader is assumed, and barcodes are required.
- For each **tape drive** contained within an ACSLS **tape library**, you must specify `acs`, `lsm`, `panel`, and `drive`. The `acs` is obtained from the tape library in which the tape drive is contained.

> **See Also:** *Oracle Secure Backup Reference* for `mkdev` syntax and semantics

E

Oracle Secure Backup and Reliable Datagram Socket (RDS)

This appendix discusses Oracle Secure Backup support for Reliable Datagram Socket (RDS). It also describes how to use RDS for communication between a client and media server.

Overview of Reliable Datagram Socket (RDS)

Reliable Datagram Socket (RDS) is an open source protocol that is used for communication over Infiniband. RDS provides a high-performance and low latency connectionless protocol for communication. It minimizes CPU utilization and is therefore preferred for communication over Infiniband.

Remote Direct Access Memory (RDMA) is a zero-copy extension of RDS. When an application performs an RDMA read or write, the application data is delivered directly to the network, thus reducing latency & enabling fast transfer. Therefore, RDMA provides high throughput. RDMA, when available, can be used with RDS for communication over Infiniband.

Using Reliable Datagram Socket (RDS) Protocol over Infiniband for Data Transfer in Oracle Secure Backup

Starting with Oracle Secure Backup 10.4, you can use the Reliable Datagram Socket (RDS) protocol over Infiniband to transfer data between a client and media server. You can also use Remote Direct Memory Access (RDMA) with RDS, thus maximizing the benefits of using RDS over Infiniband. Wherever it is possible, Oracle Secure Backup uses RDS with RDMA. When you set up an Infiniband network between a client and media server, Oracle Secure Backup automatically uses RDS to transfer data between them. If RDS is not enabled, then Oracle Secure Backup uses TCP/IP for interhost communication.

> **Note:** Oracle Secure Backup supports RDS over Infiniband for the Linux and Solaris x86 platforms. Starting with Oracle Secure Backup 10.4.0.2, RDS over Infiniband is also supported for SPARC 11.

To transfer data using RDS, both the client and media server must use Infiniband. Additionally, RDS support must be available for the operating system used by the client and media server. If the operating system does not support RDS, Oracle Secure Backup reverts to TCP/IP over Infiniband for the data transfer.

You can also set up a Preferred Network Interface (PNI) on the media server that points to the Infiniband connection.

> **See Also:** "Configuring Preferred Network Interfaces (PNI)" on page 5-8 for information about PNI

Enabling RDS for Interhost Communication

When an Infiniband connection is set up between a client and a media server, Oracle Secure Backup automatically uses RDS to transfer data between the client and media server. However, you can control the usage of RDS either at the administrative domain level or at the host level. The setting made at the host level takes precedence over the setting made at the administrative-level domain level.

Enabling RDS for the Administrative Domain

You can specify if RDS must be used for data communication between a client and media server by using one of the following interfaces:

- `obtool`

  To specify that RDS must be used for data communication, ensure that the Operations policy `disablerds` is set to `no`. This setting is applicable to the entire administrative domain. The default setting for the `disablerds` policy is `no`.

  > **See Also:** *Oracle Secure Backup Reference* for information about the `disablerds` operations policy

- Oracle Secure Backup Web tool

  In the Configure: Defaults and Policies page, select **operations** under the Policy column. On the Configure: Defaults and Policies>Operations page, ensure that the value in the Disable RDS field is set to **no** for RDS to be used.

Enabling RDS at the Host Level

For a particular host, you can specify the use of RDS by using one of the following interfaces:

- `obtool`

  To modify an existing host and enable the use of RDS for data transfer, set the `disablerds` option of the `chhost` command to `no`. During the initial configuration of a host, you can specify that RDS must be used for data transfer by setting the `disablerds` option of the `mkhost` command to `no`.

  The values you can set for the `disablerds` option are `yes`, `no`, or `systemdefault`. The default value is `systemdefault`.

  > **See Also:** *Oracle Secure Backup Reference* for information about the `disablerds` option

- Oracle Secure Backup Web tool

  Use the Disable RDS field in the Configure: Defaults and Policies>Operations page to specify the use of RDS for a particular host. To use RDS for data transfer, ensure that the Disable RDS field is set to **no**.

  The values you can select for the Disable RDS field are yes, no, or systemdefault and the default value in this field is systemdefault.

See Also: "Adding a Host to the Administrative Domain" on page 5-3 for information about disabling the use of RDS for a particular host

Glossary

active location

A **location** in a **tape library** or **tape drive**.

administrative domain

A group of computers on your network that you manage as a common unit to perform backup and restore operations. An administrative domain must include one and only one **administrative server**. It can include the following:

- One or more clients
- One or more media servers

An administrative domain can consist of a single host that assumes the **roles** of administrative server, **media server**, and **client**.

administrative server

The host that stores configuration information and **catalog** files for hosts in the **administrative domain**. There must be one and only one administrative server for each administrative domain. One administrative server can service all clients on your network. The administrative server runs the **scheduler**, which starts and monitors backups within the administrative domain.

Apache Web server

A public-domain Web server used by the Oracle Secure Backup **Web tool**.

attachment

The physical or logical connection (the path in which data travels) of a **tape device** to a host in the **administrative domain**.

automated certificate provisioning mode

A mode of **certificate** management in which the **Certification Authority (CA)** signs and then transfers **identity certificate**s to hosts over the network. This mode of issuing certificates is vulnerable to a possible, although extremely unlikely, man-in-the-middle attack. Automated mode contrasts with **manual certificate provisioning mode**.

backup encryption

The process of obscuring backup data so that it is unusable unless decrypted. Data can be encrypted at rest, in transit, or both.

backup ID

An integer that uniquely identifies a **backup section**.

backup image

The product of a backup operation. A single backup image can span multiple volumes in a **volume set**. The part of a backup image that fits on a single volume is called a **backup section**.

backup image file

The logical container of a **backup image**. A backup image consists of one file. One backup image consists of one or more **backup section**s.

backup job

A backup that is eligible for execution by the Oracle Secure Backup **scheduler**. A backup job contrasts with a **backup request**, which is an **on-demand backup** that has not yet been forwarded to the scheduler with the `backup --go` command.

backup level

The level of an **incremental backup** of file-system data. Oracle Secure Backup supports 9 different **incremental backup** levels for **file-system backup**.

backup piece

A backup file generated by **Recovery Manager (RMAN)**. A backup piece is stored in a logical container called a backup set.

backup request

An **on-demand backup** that is held locally in **obtool** until you run the `backup` command with the `--go` option. At this point Oracle Secure Backup forwards the requests to the **scheduler**, at which time each backup request becomes a **backup job** and is eligible to run.

backup schedule

A description of when and how often Oracle Secure Backup should back up the files specified by a **dataset**. The backup schedule contains the names of each **dataset file** and the name of the **media family** to use. The part of the schedule called the **trigger** defines the days and times when the backups should occur. In **obtool**, you create a backup schedule with the `mksched` command.

backup section

A portion of an **backup image file** that exists on a single tape. One **backup image** can contain one or more backup sections. Each backup section is uniquely identified by a **backup ID**.

backup transcript

A file that contains the standard output from a particular backup dispatched by the Oracle Secure Backup **scheduler**.

backup window

A time frame in which a backup operation can be run.

barcode

A symbol code, also called a tag, that is physically applied to a **volume** for identification purposes. Oracle Secure Backup supports the use of tape libraries that have an automated means to read barcodes.

blocking factor

The number of 512-byte blocks to include in each block of data written to each **tape drive**. By default, Oracle Secure Backup writes 64K blocks to tape, which is a blocking factor of 128. Because higher blocking factors usually result in better performance, you can try a blocking factor larger than the **obtar** default. If you pick a value larger than is supported by the operating system of the server, then Oracle Secure Backup fails with an error.

CA

See **Certification Authority (CA)**

catalog

A repository that records backups in an Oracle Secure Backup **administrative domain**. You can use the Oracle Secure Backup **Web tool** or **obtool** to browse the catalog and determine what files you have backed up. The catalog is stored on the **administrative server**.

certificate

A digitally signed statement from a **Certification Authority (CA)** stating that the **public key** (and possibly other information) of another entity has a value. The X.509 standard specifies the format of a certificate and the type of information contained in it: certificate version, serial number, algorithm ID, issuer, validity, subject, subject public key information, and extensions such as key usage (signing, encrypting, and so on). A variety of methods are used to encode, identify, and store the certificate.

Certification Authority (CA)

An authority in a network that performs the function of binding a **public key** pair to an identity. The CA certifies the binding by digitally signing a **certificate** that contains a representation of the identity and a corresponding public key. The **administrative server** is the CA for an Oracle Secure Backup **administrative domain**.

Certificate Revocation List (CRL)

A list used in a **public key** infrastructure that enumerates the revoked **certificate**s maintained by the **Certification Authority (CA)**.

class

A named set of **rights** for **Oracle Secure Backup user**s. A class can have multiple users, but each user can belong to one and only one class.

client

Any computer or server whose files Oracle Secure Backup backs up or restores.

content-managed expiration policy

A **volume** with this type of **expiration policy** expires when every **backup piece** on the volume is marked as deleted. You can make **Recovery Manager (RMAN)** backups, but not **file-system backup**s, to content-managed volumes. You can use RMAN to delete a **backup piece**.

cryptographic hash function

A one-way function that accepts a message as input and produces an encrypted string called a "hash" or "message digest" as output. Given the hash, it is computationally infeasible to retrieve the input. MD5 and SHA-1 are commonly used cryptographic hash functions.

cumulative incremental backup

A type of **incremental backup** in which Oracle Secure Backup copies only data that has changed at a lower **backup level**. For example, a level 3 incremental backup copies only that data that has changed since the most recent backup that is level 2 or lower.

daemons

Background processes that are assigned a task by Oracle Secure Backup during the execution of backup and restore operations. Some daemons run continually and others are started and stopped as required.

data management application (DMA)

An application that controls a backup or restore operation over the **Network Data Management Protocol (NDMP)** through connections to a **data service** and **tape service**. The DMA is the session master, whereas the NDMP services are the slaves. In an Oracle Secure Backup **administrative domain**, **obtar** is an example of a DMA.

data service

An application that runs on a client and provides **Network Data Management Protocol (NDMP)** access to database and file-system data on the primary storage system.

data transfer element (DTE)

A secondary **storage device** within a **tape library**. In tape libraries that contain multiple tape drives, data transfer elements are sequentially numbered starting with 1.

database backup storage selector

An Oracle Secure Backup configuration object that specifies characteristics of **Recovery Manager (RMAN)** SBT backups. The storage selector act as a layer between RMAN, which accesses the database, and the Oracle Secure Backup software, which manages the backup media.

dataset

The contents of a **file-system backup**. A **dataset file** describes a dataset. For example, you could create the dataset file my_data.ds to describe a dataset that includes the `/home` directory on host `brhost2`.

dataset directory

A directory that contains at least one **dataset file**. The directory groups dataset files as a set for common reference.

dataset file

A text file that describes a **dataset**. The Oracle Secure Backup dataset language provides a text-based means to define file-system data to back up.

defaults and policies

A set of configuration data that specifies how Oracle Secure Backup runs in an **administrative domain**.

device discovery

The process by which Oracle Secure Backup automatically detects devices accessed through **Network Data Management Protocol (NDMP)** and configuration changes for such devices.

attach point

A file name in the `/dev` file system on UNIX or Linux that represents a hardware **tape device**. A attach point does not specify data on disk, but identifies a hardware unit and the device driver that handles it. The inode of the file contains the device number, permissions, and ownership data. An **attachment** consists of a host name and the attach point name by which that device is accessed by Oracle Secure Backup.

differential incremental backup

A type of **incremental backup** in which Oracle Secure Backup copies only data that has changed at the same or lower **backup level**. This backup is also called a level 10 backup. Oracle Secure Backup does not support the level 10 backup on some platforms, including **Network Attached Storage (NAS)** devices such as a Network Appliance **filer**.

digital signature

A set of bits computed by an **Certification Authority (CA)** to signify the validity of specified data. The algorithm for computing the signature makes it difficult to alter the data without invalidating the signature.

DMA

See **data management application (DMA)**

domain

A group of computers and devices on a network that are administered as a unit with common rules and procedures. Within the internet, domains are defined by the IP address. All devices sharing a common part of the IP address are said to be in the same domain.

error rate

The number of recovered write errors divided by the total blocks written, multiplied by 100.

expiration policy

The means by which Oracle Secure Backup determines how a **volume** in a **media family** expires, that is, when they are eligible to be overwritten. A media family can either have a **content-managed expiration policy** or **time-managed expiration policy**.

Fiber Distributed Data Interface (FDDI)

A set of ANSI protocols for sending digital data over fiber optic cable. FDDI networks are token-passing networks, and support data rates of up to 100 Mbps. FDDI networks are typically used as backbones for wide-area networks.

Fibre Channel

A protocol used primarily among devices in a **Storage Area Network (SAN)**.

file-system backup

A backup of files on the file system initiated by Oracle Secure Backup. A file-system backup is distinct from a **Recovery Manager (RMAN)** backup made through the Oracle Secure Backup **SBT interface**.

filer

A network-attached appliance that is used for data storage.

firewall

A system designed to prevent unauthorized access to or from a private network.

full backup

An operation that backs up all of the files selected on a **client**. Unlike in an **incremental backup**, files are backed up whether they have changed since the last backup or not.

heterogeneous network

A network made up of a multitude of computers, operating systems, and applications of different types from different vendors.

host authentication

The initialization phase of a connection between two hosts in the **administrative domain**. After the hosts authenticate themselves to each other with **identity certificate**s, communications between the hosts are encrypted by **Secure Sockets Layer (SSL)**. Almost all connections are two-way authenticated; exceptions include initial host invitation to join a domain and interaction with hosts that use **NDMP access mode**.

identity certificate

An X.509 **certificate** signed by the **Certification Authority (CA)** that uniquely identifies a host in an Oracle Secure Backup **administrative domain**.

incremental backup

An operation that backs up only the files on a **client** that changed after a previous backup. Oracle Secure Backup supports 9 different incremental **backup level**s for file-system backups. A **cumulative incremental backup** copies only data that changed since the most recent backup at a lower level. A **differential incremental backup**, which is equivalent to a level 10 backup, copies data that changed since an incremental backup at the same or lower level.

An incremental backup contrasts with a **full backup**, which always backs up all files regardless of when they last changed. A full backup is equivalent to an incremental backup at level 0.

job list

A catalog created and maintained by Oracle Secure Backup that describes past, current, and pending **backup job**s.

job summary

A text file report produced by Oracle Secure Backup that describes the status of selected backup and restore jobs. Oracle Secure Backup generates the report according to a user-specified **job summary schedule**.

job summary schedule

A user-defined schedule for generating job summaries. You create job summary schedules with the `mksum` command in **obtool**.

location

A location is a place where a **volume** physically resides; it might be the name of a **tape library**, a data center, or an off-site storage facility.

logical unit number

Part of the unique identifier of a **tape device**. See **Oracle Secure Backup logical unit number** and **SCSI LUN**.

manual certificate provisioning mode

A mode of **certificate** management in which you must manually export the signed **identity certificate** for a host from the **administrative server**, transfer it to the host, and manually import the certificate into the **wallet** of the host. Unlike **automated certificate provisioning mode**, this mode is not vulnerable to a possible (if extremely unlikely) man-in-the-middle attack.

media family

A named classification of backup volumes that share the same **volume sequence file**, **expiration policy**, and **write window**.

media server

A computer or server that has at least one **tape device** connected to it. A media server is responsible for transferring data to or from the devices that are attached to it.

NAS

See **Network Attached Storage (NAS)**

native access mode

A synonym for **primary access mode**.

NDMP

See **Network Data Management Protocol (NDMP)**

NDMP access mode

The mode of access for a filer or other host that uses **Network Data Management Protocol (NDMP)** for communications within the **administrative domain**. NDMP access mode contrasts with **primary access mode**, which uses the Oracle Secure Backup network protocol. Oracle Secure Backup uses NDMP for data transfer among hosts regardless of whether a host is accessed through the primary or NDMP access modes.

Network Attached Storage (NAS)

A NAS server is a computer on a network that hosts file systems. The server exposes the file systems to its clients through one or more standard protocols, most commonly **Network File System (NFS)** and CIFS.

Network Data Management Protocol (NDMP)

An open standard protocol that defines a common architecture for backups of heterogeneous file servers on a network. This protocol allows the creation of a common agent used by the central backup application, called a **data management application (DMA)**, to back up servers running different operating systems. With NDMP, network congestion is minimized because the data path and control path are separated. Backup can occur locally—from a file server direct to a **tape drive**—while management can occur centrally.

network description file

A text file that lists the hosts in your network on which Oracle Secure Backup should be installed. For each host, you can identify the Oracle Secure Backup installation type,

the host name, and each **tape drive** attached. The install subdirectory in the **Oracle Secure Backup home** includes a sample network description file named obndf.

Network File System (NFS)

A client/server application that gives all network users access to shared files stored on computers of different types. NFS provides access to shared files through an interface called the Virtual File System (VFS) that runs on top of **TCP/IP**. Users can manipulate shared files as if they were stored on local disk. With NFS, computers connected to a network operate as clients while accessing remote files, and as servers while providing remote users access to local shared files. The NFS standards are publicly available and widely used.

OB access mode

A synonym for **primary access mode**.

obfuscated wallet

A **wallet** whose data is scrambled into a form that is extremely difficult to read if the scrambling algorithm is unknown. The wallet is read-only and is not protected by a password. An obfuscated wallet supports single sign-on (SSO).

obtar

The underlying engine of Oracle Secure Backup that moves data to and from tape. obtar is a descendent of the original Berkeley UNIX tar(2) command.

Although obtar is typically not accessed directly, you can use it to back up and restore files or directories specified on the command line. obtar enables the use of features not exposed through **obtool** or the **Web tool**.

obtool

The principal command-line interface to Oracle Secure Backup. You can use this tool to perform all Oracle Secure Backup configuration, backup and restore, maintenance, and monitoring operations. The **obtool** utility is an alternative to the Oracle Secure Backup **Web tool**.

offsite backup

A backup that is equivalent to a **full backup** except that it does not affect the full or incremental **backup schedule**. An offsite backup is useful when you want to create an **backup image** for offsite storage without disturbing your **incremental backup** schedule.

on-demand backup

A file-system backup initiated through the `backup` command in **obtool** or the Oracle Secure Backup **Web tool**. The backup is one-time-only and either runs immediately or at a specified time in the future. An on-demand backup contrasts with a **scheduled backup**, which is initiated by the Oracle Secure Backup **scheduler**.

operator

A person whose duties include backup operations, **backup schedule** management, tape swaps, and error checking.

Oracle Secure Backup home

The directory in which the Oracle Secure Backup software is installed. The Oracle Secure Backup home is typically `/usr/local/oracle/backup` on UNIX/Linux and `C:\Program Files\Oracle\Backup` on Windows. This directory contains binaries

and configuration files. The contents of the directory differ depending on which role is assigned to the host within the **administrative domain**.

Oracle Secure Backup logical unit number

A number between 0 and 31 used to generate unique attach point names during device configuration (for example, `/dev/obt0`, `/dev/obt1`, and so on). Although it is not a requirement, unit numbers typically start at 0 and increment for each additional device of a given type, whether **tape library** or **tape drive**.

The Oracle Secure Backup logical unit number is part of the name of the **attach point**. Do not confuse it with **SCSI LUN**, which is part of the hardware address of the device.

Oracle Secure Backup user

An account defined within an Oracle Secure Backup **administrative domain**. Oracle Secure Backup users exist in a separate namespace from operating system users.

overwrite

The process of replacing a file on your system by restoring a file that has the same file name.

originating location

A **location** where a **volume** was first written.

Preferred Network Interface (PNI)

The preferred network interface for transmitting data to be backed up or restored. A network can have multiple physical connections between a client and the server performing a backup or restore on behalf of that client. For example, a network can have both Ethernet and **Fiber Distributed Data Interface (FDDI)** connections between a pair of hosts. PNI enables you to specify, on a client-by-client basis, which of the server's network interfaces is preferred.

preauthorization

An optional attribute of an **Oracle Secure Backup user**. A preauthorization gives an operating system user access to specified Oracle Secure Backup resources.

primary access mode

The mode of access for a host that uses the Oracle Secure Backup network protocol for communications within the **administrative domain**. Oracle Secure Backup must be installed on hosts that use primary access mode. In contrast, hosts that use **NDMP access mode** do not require Oracle Secure Backup to be installed. Oracle Secure Backup uses **Network Data Management Protocol (NDMP)** for data transfer among hosts regardless of whether a host is accessed through the primary or NDMP access modes.

private key

A number that corresponds to a specific **public key** and is known only to the owner. Private and public keys exist in pairs in all public key cryptography systems. In a typical public key cryptosystem, such as RSA, a private key corresponds to exactly one public key. You can use private keys to compute signatures and decrypt data.

privileged backup

A file-system backup operation initiated with the `--privileged` option of the `backup` command. On UNIX and Linux systems, a privileged backup runs under the

`root` user identity. On Windows systems, the backup runs under the same account (usually `Local System`) as the Oracle Secure Backup service on the Windows client.

public key

A number associated with a particular entity intended to be known by everyone who must have trusted interactions with this entity. A public key, which is used with a corresponding **private key**, can encrypt communication and verify signatures.

Recovery Manager (RMAN)

A utility supplied with Oracle Database used for database backup, restore, and recovery. RMAN is a separate application from Oracle Secure Backup. Unlike RMAN, you can use Oracle Secure Backup to back up any file on the file system—not just database files. Oracle Secure Backup includes an **SBT interface** that RMAN can use to back up database files directly to tape.

retention period

The length of time that data in a **volume set** is not eligible to be overwritten. The retention period is an attribute of a time-managed **media family**. The retention period begins at the **write window close time**. For example, if the **write window** for a **media family** is 7 days, then a retention period of 14 days indicates that the data is eligible to be overwritten 21 days from the first write to the first **volume** in the **volume set**.

rights

Privileges within the **administrative domain** that are assigned to a **class**. For example, the `perform backup as self` right is assigned to the `operator` **class** by default. Every **Oracle Secure Backup user** that belongs to a class is granted the rights associated with this class.

roles

The functions that hosts in your network can have during backup and restore operations. There are three roles in Oracle Secure Backup: **administrative server**, **media server**, and **client**. A host in your network can serve in any of these roles or any combination of them. For example, the administrative server can also be a client and media server.

SAN

See **Storage Area Network (SAN)**

SBT interface

A media management software library that **Recovery Manager (RMAN)** can use to back up to tertiary storage. An SBT interface conforms to a published API and is supplied by a media management vendor. Oracle Secure Backup includes an SBT interface for use with RMAN.

scheduled backup

A file-system backup that is scheduled through the `mksched` command in **obtool** or the Oracle Secure Backup **Web tool** (or is modified by the `runjob` command). A **backup schedule** describes which files should be backed up. A **trigger** defined in the schedule specifies when the **backup job** should run.

scheduler

A daemon (obscheduled) that runs on an **administrative server** and is responsible for managing all backup scheduling activities. The scheduler maintains a **job list** of **backup job** operations scheduled for execution.

service daemon

A daemon (observiced) that runs on each host in the **administrative domain** that communicates through **primary access mode**. The service daemon provides a wide variety of services, including **certificate** operations.

SCSI

See **Small Computer System Interface (SCSI)**

SCSI LUN

SCSI logical unit number. A 3-bit identifier used on a SCSI bus to distinguish between up to eight devices (logical units) with the same SCSI ID. Do not confuse with **Oracle Secure Backup logical unit number**

Secure Sockets Layer (SSL)

A cryptographic protocol that provides secure network communication. SSL provides endpoint authentication through a **certificate**. Data transmitted over SSL is protected from eavesdropping, tampering or message forgery, and replay attacks.

Small Computer System Interface (SCSI)

A parallel I/O bus and protocol that permits the connection of a variety of peripherals to host computers. Connection to the SCSI bus is achieved through a host adapter and a peripheral controller.

SSL

See **Secure Sockets Layer (SSL)**

Storage Area Network (SAN)

A high-speed subnetwork of shared **storage device**s. A SAN is designed to assign data backup and restore functions to a secondary network so that they do not interfere with the functions and capabilities of the server.

storage device

A computer that contains disks for storing data.

storage element

A physical location within a **tape library** where a **volume** can be stored and retrieved by a tape library's robotic arm.

storage location

A **location** outside of a **tape library** or **tape drive** where a **volume** can be stored.

tape device

A **tape drive** or **tape library** identified by a user-defined device name.

tape drive

A **tape device** that reads and writes data stored on a tape. Tape drives are sequential-access, which means that they must read all preceding data to read any particular piece of data. The tape drives are accessible through various protocols, including **Small Computer System Interface (SCSI)** and **Fibre Channel**. A tape drive can exist standalone or in a **tape library**.

tape library

A medium changer that accepts **Small Computer System Interface (SCSI)** commands to move a **volume** from a **storage element** to a **tape drive** and back again.

tape service

A **Network Data Management Protocol (NDMP)** service that transfers data to and from secondary storage and allows the **data management application (DMA)** to manipulate and access secondary storage.

TCP/IP

Transmission Control Protocol/Internet Protocol. The suite of protocols used to connect hosts for transmitting data over networks.

three-way backup

The process of backing up an NDMP server that supports NDMP but does not have a locally attached backup device to another NDMP server that has an attached backup device. The backup is performed by sending the data through a TCP/IP connection to the NDMP server with the attached backup device. In this configuration, the NDMP data service exists on the NDMP server that contains the data to be backed up and the NDMP tape service exists on the NDMP server with the attached tape device.

time-managed expiration policy

A **media family expiration policy** in which every **volume** in a **volume set** can be overwritten when it reaches its **volume expiration time**. Oracle Secure Backup computes the volume expiration time by adding the **volume creation time** for the first volume in the set, the **write window time**, and the **retention period**.

For example, you set the write window for a **media family** to 7 days and the **retention period** to 14 days. Assume that Oracle Secure Backup first wrote to the first volume in the set on January 1 at noon and subsequently wrote data on 20 more volumes in the set. In this scenario, all 21 volumes in the set expire on January 22 at noon.

You can make a **Recovery Manager (RMAN)** backup or a **file-system backup** to a **volume** that use a time-managed expiration policy.

trigger

The part of a **backup schedule** that specifies the days and times at which the backups should occur.

trusted certificate

A **certificate** that is considered valid without validation testing. Trusted certificates build the foundation of the system of trust. Typically, they are certificates from a trusted **Certification Authority (CA)**.

unprivileged backup

File-system backups created with the `--unprivileged` option of the `backup` command. When you create or modify an **Oracle Secure Backup user**, you associate operating system accounts with this user. Unprivileged backups of a host run under the operating system account associate with Oracle Secure Backup user who initiates the backup.

volume

A volume is a unit of media, such as an 8mm tape. A volume can contain multiple backup images.

volume creation time

The time at which Oracle Secure Backup wrote **backup image** file number 1 to a **volume**.

volume expiration time

The date and time on which a **volume** in a **volume set** expires. Oracle Secure Backup computes this time by adding the **write window** duration, if any, to the **volume creation time** for the first volume in the set, then adding the volume **retention period**.

For example, assume that a volume set belongs to a **media family** with a retention period of 14 days and a write window of 7 days. Assume that the **volume creation time** for the first volume in the set was January 1 at noon and that Oracle Secure Backup subsequently wrote data on 20 more volumes in the set. In this scenario, the volume expiration time for all 21 volumes in the set is January 22 at noon.

volume ID

A unique alphanumeric identifier assigned by Oracle Secure Backup to a **volume** when it was labeled. The volume ID usually includes the **media family** name of the **volume**, a dash, and a unique **volume sequence number**. For example, a volume ID in the RMAN-DEFAULT media family could be RMAN-DEFAULT-000002.

volume label

The first block of the first **backup image** on a **volume**. It contains the **volume ID**, the owner's name, the **volume creation time**, and other information.

volume sequence file

A file that contains a unique **volume ID** to assign when labeling a **volume**.

volume sequence number

A number recorded in the **volume label** that indicates the order of volumes in a **volume set**. The first **volume** in a set has sequence number 1. The **volume ID** for a volume usually includes the **media family** name of the volume, a dash, and a unique volume sequence number. For example, a volume ID for a volume in the RMAN-DEFAULT media family could be RMAN-DEFAULT-000002.

volume set

A group of volumes spanned by a **backup image**. The part of the backup image that fits on a single **volume** is a **backup section**.

volume tag

A field that is commonly used to hold the **barcode** identifier, also called a volume tag, for the **volume**. The volume tag is found in the **volume label**.

wallet

A password-protected encrypted file. An Oracle wallet is primarily designed to store X.509 certificates and their associated **public key**/**private key** pair. The contents of the wallet are only available after the wallet password has been supplied, although with an **obfuscated wallet** no password is required.

Web tool

The browser-based GUI that enables you to configure an **administrative domain**, manage backup and restore operations, and browse the backup **catalog**.

write window

The period for which a **volume set** remains open for updates, usually by appending an additional **backup image**. The write window opens at the **volume creation time** for the first **volume** in the set and closes after the write window period has elapsed. After the **write window close time**, Oracle Secure Backup does not allow further updates to the volume set until it expires (as determined by its **expiration policy**), or until it is relabeled, reused, unlabeled, or forcibly overwritten.

A write window is associated with a **media family**. All volume sets that are members of the media family remain open for updates for the same time period.

write window close time

The date and time that a **volume set** closes for updates. Oracle Secure Backup computes this time when it writes **backup image file** number 1 to the first **volume** in the set. If a volume set has a **write window close time**, then this information is located in the volume section of the **volume label**.

write window time

The length of time during which writing to a **volume set** is permitted.

Index

A

B

C

D

E

F

H

I

K

L

M

N

O

P

R

S

T

U

V

W

X