

Oracle® Secure Backup

Installation and Configuration Guide

Release 10.2

E05408-02

July 2008

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

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, Janet Stern, Radhika Vullikanti, Joe Wadleigh, Steve Wertheimer

The Programs (which include both the software and documentation) contain proprietary information; they are provided under a license agreement containing restrictions on use and disclosure and are also protected by copyright, patent, and other intellectual and industrial property laws. Reverse engineering, disassembly, or decompilation of the Programs, except to the extent required to obtain interoperability with other independently created software or as specified by law, is prohibited.

The information contained in this document is subject to change without notice. If you find any problems in the documentation, please report them to us in writing. This document is not warranted to be error-free. Except as may be expressly permitted in your license agreement for these Programs, no part of these Programs may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose.

If the Programs are delivered to the United States Government or anyone licensing or using the Programs on behalf of the United States Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the Programs, including documentation and technical data, shall be subject to the licensing restrictions set forth in the applicable Oracle license agreement, and, to the extent applicable, the additional rights set forth in FAR 52.227-19, Commercial Computer Software—Restricted Rights (June 1987). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

The Programs are not intended for use in any nuclear, aviation, mass transit, medical, or other inherently dangerous applications. It shall be the licensee's responsibility to take all appropriate fail-safe, backup, redundancy and other measures to ensure the safe use of such applications if the Programs are used for such purposes, and we disclaim liability for any damages caused by such use of the Programs.

Oracle, JD Edwards, PeopleSoft, and Siebel are registered trademarks of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

The Programs may provide links to Web sites and access to content, products, and services from third parties. Oracle is not responsible for the availability of, or any content provided on, third-party Web sites. You bear all risks associated with the use of such content. If you choose to purchase any products or services from a third party, the relationship is directly between you and the third party. Oracle is not responsible for: (a) the quality of third-party products or services; or (b) fulfilling any of the terms of the agreement with the third party, including delivery of products or services and warranty obligations related to purchased products or services. Oracle is not responsible for any loss or damage of any sort that you may incur from dealing with any third party.

Contents

Preface	ix
Audience	ix
Documentation Accessibility	ix
Related Documents	x
Conventions	x
1 Introduction to Oracle Secure Backup	
What Is Oracle Secure Backup?	1-1
Oracle Secure Backup Concepts	1-2
Oracle Secure Backup Administrative Domains and Hosts	1-2
Host Roles in an Administrative Domain	1-2
Host Naming in an Administrative Domain	1-3
Oracle Secure Backup Host Access Modes	1-3
Oracle Secure Backup Administrative Domain: Examples	1-4
Tape Devices	1-5
Tape Drives	1-5
Tape Libraries	1-7
Virtual Tape Libraries	1-9
Device Names and Attachments	1-10
Oracle Secure Backup Interfaces	1-10
System Requirements for Oracle Secure Backup	1-11
Disk Space Requirements for Oracle Secure Backup	1-11
Other System Requirements for Oracle Secure Backup	1-12
Linux Media Server System Requirement: SCSI Generic Driver	1-12
Acquiring Oracle Secure Backup Installation Media	1-12
Installation and Configuration Overview	1-13
About Upgrade Installations	1-14
Preparing Administrative Domain Hosts for Upgrade to Release 10.2.0.2	1-14
2 Installing Oracle Secure Backup on Linux or UNIX	
Oracle Secure Backup Linux and UNIX Installation Steps	2-1
Prerequisites for Installing Oracle Secure Backup on Linux and UNIX	2-2
Prerequisites for Installation on Linux	2-2
Required SCSI Tape Device Parameters on Linux and UNIX	2-3
Assigning Oracle Secure Backup Logical Unit Numbers to Devices	2-3

Extracting Oracle Secure Backup from OTN Download on Linux or UNIX	2-4
Preparing to Install Oracle Secure Backup on Linux and UNIX.....	2-5
Creating the Oracle Secure Backup Home	2-5
Loading Oracle Secure Backup Software on Linux or UNIX Using setup Script.....	2-5
Configuring Installation Parameters in the obparameters File.....	2-7
Installing Oracle Secure Backup on Linux or UNIX with installob.....	2-8
Installing or Uninstalling Oracle Secure Backup on AIX	2-12
Running installob Again to Create Device Special Files	2-12
Creating Device Special Files with makedev	2-12
Linux Attach Points	2-14
Upgrade Installation on Linux or UNIX	2-14
Uninstalling Oracle Secure Backup on Linux or UNIX.....	2-15

3 Installing Oracle Secure Backup on Windows

Preliminary Steps	3-1
Disabling Removable Storage Service on Windows Media Servers	3-2
Extracting Oracle Secure Backup from OTN Download on Windows.....	3-2
Running the Oracle Secure Backup Windows Installer	3-3
Configuring Oracle Secure Backup	3-13
Configuring Firewalls for Oracle Secure Backup on Windows.....	3-17
Upgrade Installation on Windows 32-Bit	3-17
Upgrade Installation on Windows x64.....	3-18
Uninstalling Oracle Secure Backup on Windows.....	3-19

4 Oracle Secure Backup User Interfaces

Using Oracle Secure Backup in Enterprise Manager	4-1
Enabling Oracle Secure Backup Links in Oracle Enterprise Manager	4-2
Registering an Administrative Server in Oracle Enterprise Manager.....	4-3
Accessing the Web Tool from Enterprise Manager.....	4-3
Using the Oracle Secure Backup Web Tool	4-4
Starting a Web Tool Session	4-4
Web Tool Home Page	4-5
Persistent Page Links	4-6
Web Tool Configure Page	4-7
Web Tool Manage Page.....	4-8
Web Tool Backup Page.....	4-10
Web Tool Restore Page.....	4-10
Using obtool	4-11
Displaying Help for Invoking obtool	4-11
Starting obtool in Interactive Mode.....	4-11
Running obtool Commands in Interactive Mode.....	4-12
Redirecting obtool Input from Text Files	4-12
Executing obtool Commands in Noninteractive Mode.....	4-12
Running Multiple Commands in Noninteractive Mode.....	4-12
Redirecting Input in Noninteractive Mode.....	4-13
Ending an obtool Session	4-13
Starting obtool as a Specific User.....	4-13

5 Administrative Domain Configuration and Management

Administrative Domain Configuration Overview	5-1
Administrative Domain Configuration Steps: Outline.....	5-1
Configuring the Administrative Domain with Hosts.....	5-2
About Administrative Domain Host Configuration.....	5-2
Viewing the Hosts in the Administrative Domain.....	5-3
Adding a Host to the Administrative Domain	5-4
Adding the Media Server Role to an Administrative Server.....	5-6
Adding Backup and Restore Environment Variables to an NDMP Host.....	5-8
Configuring Preferred Network Interfaces (PNI)	5-8
Pinging a Host	5-9
Viewing or Editing Host Properties	5-9
Updating a Host	5-10
Removing a Host.....	5-10
Adding Tape Devices to an Administrative Domain	5-11
Tape Device Names	5-11
About Configuring Tape Drives and Libraries.....	5-11
Updating a Tape Device Inventory	5-13
Displaying the Devices Page	5-14
Configuring a Tape Library	5-15
Configuring Automatic Tape Cleaning for a Library	5-17
Configuring a Tape Drive	5-17
Automatic Discovery of Tape Devices on NDMP Hosts.....	5-19
Adding a Tape Device Attachment	5-20
Pinging a Device Attachment.....	5-21
Displaying Device Attachment Properties.....	5-22
Multiple Attachments for SAN-Attached Tape Devices	5-22
Creating Attach Points for Solaris 10 Fibre Channel Devices.....	5-22
Pinging a Tape Device.....	5-24
Displaying Device Properties.....	5-24
Editing Device Properties	5-24

6 Security for Backup Networks

Backup Network Security Overview.....	6-1
Planning Security for an Administrative Domain.....	6-2
Identifying Assets and Principals.....	6-2
Identifying Your Backup Environment Type	6-3
Single System.....	6-3
Data Center	6-4
Corporate Network.....	6-6
Choosing Secure Hosts for the Administrative and Media Servers	6-6
Determining the Distribution Method of Host Identity Certificates	6-7
Trusted Hosts.....	6-8
Host Authentication and Communication	6-9
Identity Certificates and Public Key Cryptography	6-9
Authenticated SSL Connections.....	6-10

Certification Authority	6-10
Automated and Manual Certificate Provisioning Mode	6-11
Oracle Wallet.....	6-11
Oracle Secure Backup Encryption Wallet.....	6-12
Web Server Authentication.....	6-13
Revoking a Host Identity Certificate.....	6-13
Encryption of Data in Transit	6-14
Default Security Configuration	6-15
Configuring Security for the Administrative Domain	6-16
Providing Certificates for Hosts in the Administrative Domain	6-16
Configuring the Administrative Server.....	6-16
Configuring Media Servers and Clients	6-17
Setting the Size for Public and Private Keys	6-18
Setting the Key Size in obparameters.....	6-19
Setting the Key Size in the certkeysize Security Policy	6-19
Setting the Key Size in mkhost.....	6-20
Enabling and Disabling SSL for Host Authentication and Communication.....	6-20
Managing Certificates with obcm	6-21
Exporting Signed Certificates.....	6-21
Importing Signed Certificates	6-21

A Oracle Secure Backup Directories and Files

Oracle Secure Backup Home Directory	A-1
Administrative Server Directories and Files.....	A-1
Media Server Directories and Files.....	A-4
Client Host Directories and Files	A-5

B Oracle Secure Backup obparameters Installation Parameters

customized obparameters	B-1
start daemons at boot	B-2
identity certificate key size.....	B-2
create preauthorized oracle user.....	B-2
default UNIX user	B-3
default UNIX group	B-3
linux ob dir and solaris64 ob dir	B-3
linux db dir and solaris64 db dir.....	B-4
linux temp dir and solaris64 temp dir	B-4
linux links and solaris64 links	B-4
ask about ob dir	B-5
default protection	B-5
run obopenssl.....	B-6

C Determining Linux and UNIX SCSI Parameters

Determining SCSI Device Parameters on Linux.....	C-1
Determining SCSI Device Parameters on Solaris.....	C-2
Probing SCSI Target ID and LUN for Media Devices From Solaris Open Boot PROM	C-2

Viewing SCSI Bus Name-Instance Parameter Values in Solaris	C-4
D Manually Configuring Solaris Drivers	
Installing the Oracle Secure Backup Device Driver Manually	D-1
Uninstalling the Oracle Secure Backup Device Driver Manually	D-2
E Oracle Secure Backup and ACSLS	
About ACSLS	E-1
ACSLs and Oracle Secure Backup	E-2
Communicating with ACSLS	E-3
Drive Association	E-3
Volume Loading and Unloading	E-3
Imports and Exports	E-4
Access Controls	E-4
Scratch Pool Management	E-4
Modified Oracle Secure Backup Commands	E-4
Unsupported Oracle Secure Backup Commands	E-5
Installation and Configuration	E-5
Index	

Preface

This Preface contains these topics:

- [Audience](#)
- [Documentation Accessibility](#)
- [Related Documents](#)
- [Conventions](#)

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 Oracle Secure Backup and Recovery and Recovery Manager concepts.

Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible, with good usability, to the disabled community. To that end, our documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. Accessibility standards will continue to evolve over time, and Oracle is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be accessible to all of our customers. For more information, visit the Oracle Accessibility Program Web site at

<http://www.oracle.com/accessibility/>

Accessibility of Code Examples in Documentation

Screen readers may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an otherwise empty line; however, some screen readers may not always read a line of text that consists solely of a bracket or brace.

Accessibility of Links to External Web Sites in Documentation

This documentation may contain links to Web sites of other companies or organizations that Oracle does not own or control. Oracle neither evaluates nor makes any representations regarding the accessibility of these Web sites.

TTY Access to Oracle Support Services

Oracle provides dedicated Text Telephone (TTY) access to Oracle Support Services within the United States of America 24 hours a day, 7 days a week. For TTY support, call 800.446.2398. Outside the United States, call +1.407.458.2479.

Related Documents

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

- *Oracle Secure Backup Migration Guide*

This book explains how to migrate from Reliaty Backup to Oracle Secure Backup.

- *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/technology/products/secure-backup>

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

<http://www.oracle.com/technology/software>

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.
<i>italic</i>	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.

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 **Storage Area Network (SAN)**, Gigabit Ethernet (GbE), and **Small Computer System Interface (SCSI)** environments using standard tape formats.

Using Oracle Secure Backup on your network enables you to take data from a networked host running Oracle Secure Backup or a **Network Attached Storage (NAS)** device that support **Network Data Management Protocol (NDMP)**, and back up that data on a **tape device** attached to your 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 in conjunction 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 Administrative Domain: Examples](#)
- [Tape Devices](#)

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 [roles](#):

- **Administrative server**

Each administrative domain must have exactly one [administrative server](#). During post-installation 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 [tape library](#). It might have attachments to multiple tape libraries.

You specify the attachments between media servers and tape devices during post-installation 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 **Network Data Management Protocol (NDMP)** can also serve the client role.

Note: A host can be assigned more than one role 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 **Network Data Management Protocol (NDMP)**, but there are variations among implementations and versions of NDMP. 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, Mirapoint, and DinoStor 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 2, 3, 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 server also runs on the same host.

Figure 1–1 Administrative Domain with One Host

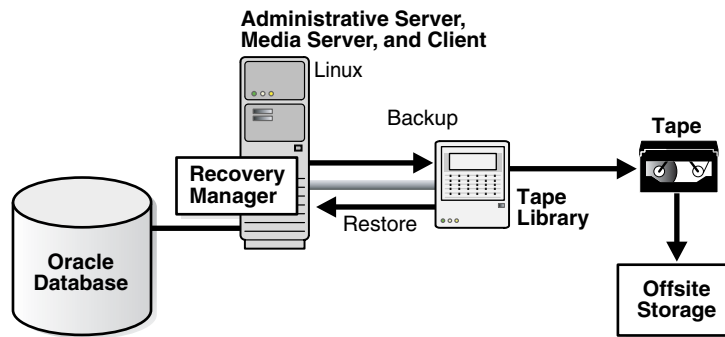
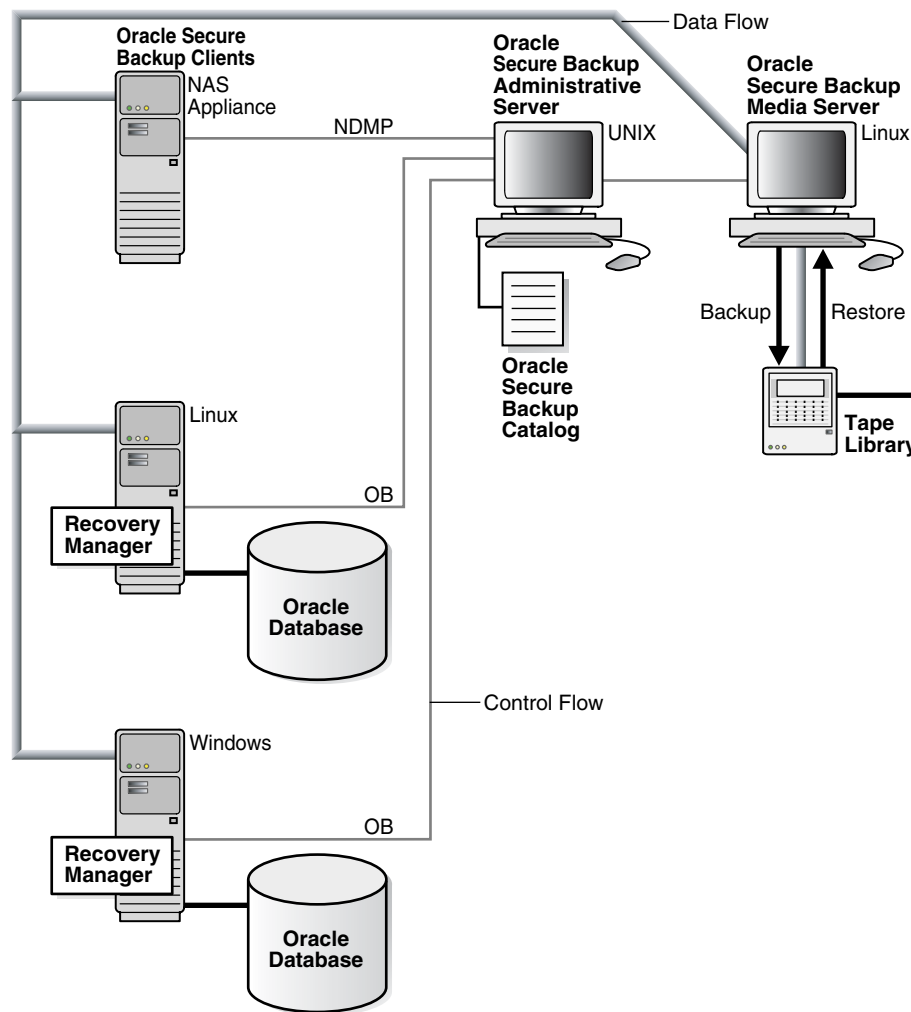


Figure 1–2 shows a possible Oracle Secure Backup administrative domain that includes three client hosts, one administrative server, and one media server. A **Network Attached Storage (NAS)** appliance contains ordinary file data. One client based on UNIX and another based on Windows contain databases, as well as other file data. Oracle Secure Backup can back up to tape the non-database files on file systems accessible on client hosts. **Recovery Manager (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 that are 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 just 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 that are contained in that block. So, for example, the Oracle Secure Backup default blocking factor (128) results in a tape block size of 128*512 bytes or 64KB.

The *maximum blocking factor* is an upper limit on the blocking factor that Oracle Secure Backup will use. 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 will be 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 and/or maximum blocking factor 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 and/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) in effect as a result of applying 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 OS will have 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 need to reposition the tape, 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. This means that the maximum blocking factor must always be greater than or equal to the largest block size you will 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. Common tape formats include the following:

- 8mm
- 4mm, or Digital Audio Tape (DAT)
- Advanced Intelligent Tape (AIT)
- Digital Data Storage (DDS)
- Digital Linear Tape (DLT) and Super DLT (SDLT)
- Linear Tape-Open (LTO), an open alternative to the proprietary DLT format

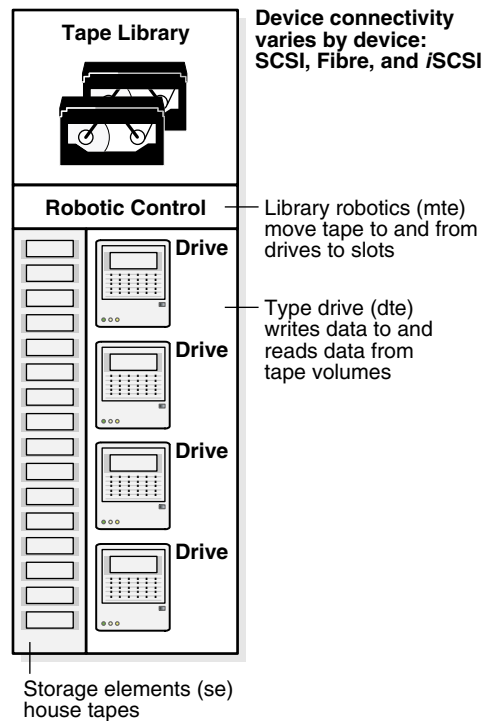
Information about the tape formats of tape devices supported by Oracle Secure Backup is available at the following URL:

<http://www.oracle.com/technology/products/secure-backup>

Tape Libraries

A **tape library** is a robotic **tape device** that accepts **Small Computer System Interface (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, a number of 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 consists of a set of addressable elements, each of which can contain a tape or can be used to move a tape. 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 tape library. After the door is closed, the robotic arm transfers cartridges to internal slots in the tape library. Because the tape library itself is not opened during this procedure, a re-inventory is not 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 there is more than one element of a particular type, element numbering starts at 1. When there is only one element of a type, the number can be omitted: **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 a number of 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 as well as 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. Each **tape device**, therefore, 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 is connected 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 itself. An attachment usually includes the identity of a host plus a **device special file** name in Linux or UNIX, a device name in Windows, or a **Network Attached Storage (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.
- 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 the case of **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 Metalink, at the following URL:

<http://metalink.oracle.com/>

Information about every [tape device](#) supported by Oracle Secure Backup is available at the following URL:

<http://www.oracle.com/technology/products/secure-backup/>

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 the disk space requirements for installing an [administrative server](#), [media server](#), or [client](#) on Linux and UNIX.

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

Oracle Secure Backup Installation and Packages	Disk Space
Administrative server for UNIX installation (can include the media server, client, or both)	60 MB
Administrative server for Linux installation (can include the media server, client, or both)	40 MB
Common administrative server files for all operating systems	10 MB
Media server, client, or both	50 MB

[Table 1–2](#) describes the 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)	20 MB
Media server, client, or both (no administrative server)	15 MB

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

Other System Requirements for Oracle Secure Backup

Each host that participates in a Oracle Secure Backup **administrative domain** must run **TCP/IP (Transmission Control Protocol/Internet Protocol)**. 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 one of the versions 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/technology/products/secure-backup>

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 will 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 and UNIX 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, "Administrative Domain Configuration and Management"](#). 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 10.2.0.2, then you must upgrade every host in the Oracle Secure Backup administrative domain. Oracle Secure Backup release 10.2 is incompatible with Oracle Secure Backup release 10.1.

If you are upgrading an existing Oracle Secure Backup release 10.2.0.1 installation to Oracle Secure Backup release 10.2.0.2, then you must upgrade the administrative server and media server hosts in the Oracle Secure Backup administrative domain. Oracle recommends that you also update your clients in the domain to Oracle Secure Backup release 10.2.0.2. But hosts acting as clients only can remain at Oracle Secure Backup release 10.2.0.1 and participate as part of an Oracle Secure Backup release 10.2.0.2 domain.

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 10.2.0.2, 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 10.2.0.2

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 methods for stopping the Oracle Secure Backup daemons are as follows.

On Linux:

```
# /etc/init.d/observed stop
```


On Solaris:

```
#/etc/init.d/OracleSecureBackup stop
```

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

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

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

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

1. Open the Services applet.
2. Right-click the **Oracle Secure Backup Services** service, and select **Stop**.

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 these sections:

- [Oracle Secure Backup Linux and UNIX Installation Steps](#)
- [Prerequisites for Installing Oracle Secure Backup on Linux and UNIX](#)
- [Extracting Oracle Secure Backup from OTN Download on Linux or UNIX](#)
- [Preparing to Install Oracle Secure Backup on Linux and UNIX](#)
- [Creating the Oracle Secure Backup Home](#)
- [Loading Oracle Secure Backup Software on Linux or UNIX Using setup Script](#)
- [Configuring Installation Parameters in the obparameters File](#)
- [Installing Oracle Secure Backup on Linux or UNIX with installob](#)
- [Installing or Uninstalling Oracle Secure Backup on AIX](#)
- [Running installob Again to Create Device Special Files](#)
- [Creating Device Special Files with makedev](#)
- [Upgrade Installation on Linux or UNIX](#)
- [Uninstalling Oracle Secure Backup on Linux or UNIX](#)

Oracle Secure Backup Linux and UNIX Installation Steps

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

- **Loading**
Files required for installing Oracle Secure Backup on one or more different Linux or UNIX platforms are staged on the [administrative server](#), in a directory called the [Oracle Secure Backup home](#). This step is performed by a script called setup.
- **Installing**
Oracle Secure Backup executables are deployed correctly for use on the host. This step is performed by a script called installob.

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

- Device special file creation 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 (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 (Transmission Control Protocol/Internet Protocol)**.
- 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 **Small Computer System Interface (SCSI)** parameters for each **tape drive** and **tape library** attached to your Linux or UNIX **media server**. You can determine them using the procedures described in [Appendix C, "Determining Linux and UNIX SCSI Parameters"](#). You need this information when creating a **device special file** for each **tape device**.
- 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. It 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 should be used, 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 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 **Small Computer System Interface (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**.

On Linux the required SCSI parameters are:

- The host bus adapter number for the SCSI adapter
- The SCSI bus address
- The SCSI target ID
- The **SCSI logical unit number (SCSI LUN)**

Note: In the Linux operating system, SCSI bus addresses are referred to as channels.

On Solaris the required SCSI parameters are:

- The SCSI bus name-instance
- The SCSI target ID
- The SCSI LUN

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 logical unit number with the **Oracle Secure Backup logical unit number**. The Oracle Secure Backup logical unit number is a number you assign that is used in generating a **device special file** name.

See Also: [Appendix C, "Determining Linux and UNIX SCSI Parameters"](#) for details on how to collect this information

Assigning Oracle Secure Backup Logical Unit Numbers to Devices

Each **tape drive** and **tape library** must be assigned an **Oracle Secure Backup logical unit number** 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: "[Linux Attach Points](#)" on page 2-14

Note: The Oracle Secure Backup logical unit number should not be confused with the **SCSI logical unit number (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 filename.

Extracting Oracle Secure Backup from OTN Download on Linux or UNIX

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/technology/products/secure-backup>

4. Click **Free Download**.

The Oracle Technology Network Developer License Terms page appears.

5. Read **Export Controls on the Programs** and select the **Yes, I accept...** option.

Read the **Oracle Technology Network Development License Agreement** and click **I Accept**.

The Oracle Secure Backup Downloads page appears.

6. Select the **Accept License Agreement** option, and click the link for the version of Oracle Secure Backup release 10.2 specific to your operating system.

Note: If you have more than one operating system in your environment, then you must perform multiple downloads of the Oracle Secure Backup release 10.2 software.

7. The Oracle Secure Backup release 10.2 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 10.2.

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 **Small Computer System Interface (SCSI)** parameters for each **tape drive** and **tape library** attached to your Linux and UNIX media servers. You need this information when creating **device special file** 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 has to 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 Real Application Clusters (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** on each host before beginning the process of loading and installing the software. 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.

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 is used to perform the loading process, in which packages of files required to install Oracle Secure Backup on one or more platforms 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 the setup program as follows:

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

The installation files are compressed, and Oracle Secure Backup expands them in a temporary directory during installation. If you want to specify a particular 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 process:

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

The setup script displays a welcome message and tells you what version of Oracle Secure Backup it contains. It then lists all packages available and asks you to enter a space-separated list of package numbers you want. If you want to load all packages, then enter `all`.

4. Each package contains the files required to install Oracle Secure Backup on that platform. Enter the number or numbers that identify the installation packages that you want to load, so that they will be staged for later installation on Linux or UNIX hosts in your [administrative domain](#).

Note: You can run setup again in the future if necessary, to load packages for more platforms into the Oracle Secure Backup home, without affecting existing installations of Oracle Secure Backup.

The setup script displays progress messages as it loads the package or packages you selected. When it is finished, it tells 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 on the platforms you specified are stored in the Oracle Secure Backup home on this host.

5. At this point the setup script can start the `installob` script to install Oracle Secure Backup on the local host. You can also defer this task until later.

Choose one of the following:

- Enter `no` if you want 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 the setup script 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 it is possible that some temporary files, named OBnnnn or OBnnnn.Z, 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, username and password) that are valid at the **client** with required privileges after you complete the Oracle Secure Backup installation. Otherwise, Oracle Secure Backup is unable to 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. Start the installob script.

The Oracle Secure Backup setup script ends by asking if you want 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 the installob script from the command prompt. While logged in as `root`, go to the **Oracle Secure Backup home** and enter the following command:

```
install/installob
```

Note: Before starting the installob script on a **media server**, have the **SCSI** parameters for each **tape device** available. You will have the option of entering those parameters to create a **device special file** for each **Small Computer System Interface (SCSI)** device as part of the initial installation.

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**.

2. 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 already 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 3.

- 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 the following:
 - Enter yes or press the Enter key to indicate that you do not want to customize the obparameters file. Continue to step 3.
 - Enter yes 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.

3. Specify the host role.

You determined the [roles](#) for each host when planning your [administrative domain](#). Choose one of the following:

- 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. But 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 device special files 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.

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

The installob script asks for a password for the admin [Oracle Secure Backup user](#), and then asks you to re-enter 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.

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. Create a password for the Oracle Secure Backup keystore.

The installob script asks for a password for the Oracle Secure Backup keystore, and then asks you to re-enter 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.

6. Enter an e-mail address for notifications.

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

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.

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

Note: In installob, the term *configuring* refers to creating the device special files 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, "Administrative Domain Configuration and Management"](#).

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 one of the following:

- Enter no if you do not want to create device special files for your tape devices at this time, or if you are installing on an administrative server with no tape devices attached.

Note: On Linux systems Oracle recommends that you enter no when asked if you want to configure tape libraries or tape drives during installation. The preferred mode of operation on Linux systems is to use the /dev/sg devices for attach points, as described in ["Linux Attach Points"](#) on page 2-14.

If you choose to create device special files later, or if you add a tape device to a media server in the future, then see ["Running installob Again to Create Device Special Files"](#) on page 2-12 or ["Creating Device Special Files with makedev"](#) on page 2-12 for two alternative methods of completing this task.

- Enter **yes** to configure tape devices now.

To create device special files, the installob script asks if there are tape libraries 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 completed entering information about tape libraries attached to this host, the installob script asks the same questions about standalone tape drives.

The SCSI parameters requested by the installob script differ somewhat between Linux and UNIX hosts. On Linux hosts, the installob script asks for:

- Oracle Secure Backup LUN
- Host bus adapter
- SCSI bus address
- Target ID
- **SCSI logical unit number (SCSI LUN)**

Note: In the Linux operating system, SCSI bus addresses are referred to as channels.

On UNIX hosts, the installob script asks for:

- Oracle Secure Backup LUN
- SCSI bus name-instance
- SCSI Target ID
- SCSI LUN

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 filename is needed when you configure the **attachment** for the tape device, as part of configuring the Oracle Secure Backup domain. The filename should be `/dev/obtn` for tape drives, and `/dev/obltn` 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, as described in ["Running installob Again to Create Device Special Files"](#) on page 2-12, entering the correct values, or use the makedev script described in ["Creating Device Special Files with makedev"](#) on page 2-12.

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

8. 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 current release of Oracle Secure Backup supports only the client role on an AIX platform. The Oracle Secure Backup Administrative Server and Media Server roles are not supported. The previous version of Oracle Secure Backup must be uninstalled prior to installing the current release of Oracle Secure Backup.

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

See Also:

- ["Installing Oracle Secure Backup on Linux or UNIX with `installob`"](#) on page 2-8
- ["Uninstalling Oracle Secure Backup on Linux or UNIX"](#) on page 2-15

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: This mapping of the Oracle Secure Backup `admin` user can be changed post-installation.

Running `installob` Again to Create Device Special Files

You can run the `installob` script again on a host on which Oracle Secure Backup is already installed. The chief reason to do so is to create device special files for several additional tape libraries or drives on a [media server](#), instead of invoking `makedev` repeatedly to create one [device special file](#) at a time.

If you run the `installob` script on a host on which it has already been run, then the script detects the existing installation and asks if you want to perform each installation stage again. You can skip tasks that were correctly completed before by entering `no` when asked whether you want to perform them again.

You can then continue to enter the device special file configuration information, as described in ["Installing Oracle Secure Backup on Linux or UNIX with `installob`"](#) on page 2-8.

Creating Device Special Files with `makedev`

The `makedev` script in Oracle Secure Backup is used to create a [device special file](#) for a single [tape device](#). Internally, the `installob` script calls `makedev` once for each tape

device specified during installation. You can also run makedev directly from the command line if you must create device special files for a single tape device.

The makedev script can also replace an old device special file, rather than adding a new one. If you reuse an **Oracle Secure Backup logical unit number** for a **tape library** or **tape drive**, then the device special files for the old tape device are overwritten.

If you must create device special files for several tape devices, then it might be more convenient to use the installob script as described in "[Running installob Again to Create Device Special Files](#)" on page 2-12.

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

Note: On a Linux or other UNIX host, the parameters prompted for are different.

To use the makedev script to create device special files for a tape device:

1. Log on as root.
2. Go to the install directory in your **Oracle Secure Backup home**. For example:

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

3. Enter the makedev command at the shell prompt:

```
# makedev
```

4. The makedev script asks for information about the tape device.

The **Small Computer System Interface (SCSI)** parameters requested by the script differ somewhat between Linux and UNIX hosts. On Linux hosts, the script asks for:

- Oracle Secure Backup logical unit number

Do not confuse the Oracle Secure Backup logical unit number with the **SCSI logical unit number (SCSI LUN)**.

- Device type

Enter d for a tape drive or l (lowercase L) for a tape library.

- Host bus adapter
- SCSI bus address
- Target ID
- SCSI logical unit number

Note: In the Linux operating system, SCSI bus addresses are referred to as channels.

On UNIX hosts, the makedev script asks for:

- Oracle Secure Backup logical unit number

Do not confuse the Oracle Secure Backup logical unit number with the SCSI logical unit number.

- Device type

Enter `d` for a tape drive or `l` (lowercase L) for a tape library.

- SCSI bus name-instance
 - SCSI target ID
 - SCSI logical unit number
5. The `makedev` script creates the device special file, displaying messages indicating its progress. When it is finished, the `makedev` script displays the name of the new device special file and exits.

Linux Attach Points

Oracle recommends that for Linux systems the `/dev/sg` devices be used as attach points with Oracle Secure Backup. The use of the Oracle Secure Backup `/dev/ob` devices has certain limitations that some users might find unacceptable in their environments. For example the logical unit number cannot be greater than 7, and the **Small Computer System Interface (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. The following example shows a system that has five tape drives and four tape libraries available:

```
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
```

After you have identified the `/dev/sg` that corresponds to the tape device you are interested in, you can use the `obtool mkdev` command to create attach points for it. For example, to create the attach points for an ADIC Fastor 2 **tape library** that has one **tape drive**, you would enter the following:

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

Upgrade Installation on Linux or UNIX

To upgrade a Linux or UNIX installation of Oracle Secure Backup, follow the setup and installation process described in ["Installing Oracle Secure Backup on Linux or UNIX with installob"](#) on page 2-8. During the upgrade process, the installer displays the following prompt:

```
Oracle Secure Backup is already installed on this machine (myhostname-sun2).
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. Shut down processes related to Oracle Secure Backup, such as the http processes for the Oracle Secure Backup [Web tool](#). Use the following command to identify processes related to Oracle Secure Backup:

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

Use the following command to kill each process in the list associated with Oracle Secure Backup, where *pid* is the process ID:

```
kill -9 pid
```

3. Change directory to the parent directory of the [Oracle Secure Backup home](#) directory. For example:

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

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

4. Run the `uninstallob` script:

```
# backup/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.

5. Enter the name of a host from which you want to uninstall Oracle Secure Backup.
6. 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`.

7. The `uninstallob` script asks if you want to remove the [Oracle Secure Backup home](#) directory. Select one of the following:

- `no`

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

- `yes`

Select this option if you want to remove the Oracle Secure Backup home directory. All files in the home directory will be 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.

8. The `uninstallob` script asks if you want 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 the following:
 - 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.
9. The `uninstallob` script displays the choices you have made and asks if you want to continue with the uninstallation on this host. Select one of the following:
 - 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.
10. If you entered `no` in the previous step, then the `uninstallob` script asks if you want to uninstall Oracle Secure Backup from another host. Select one of the following:
 - yes
If you select this option, then the `uninstallob` script repeats steps 5 through 9 for a different host
 - no
If you select this option, then the `uninstallob` script exits.

Note:

- If you uninstall Oracle Secure Backup from the local computer, then the `uninstallob` script removes the directory `/usr/local/oracle/backup` when it completes.
 - On UNIX it might be necessary to remove the driver for Oracle Secure Backup manually from each **media server** after uninstalling the rest of the product. See "[Uninstalling the Oracle Secure Backup Device Driver Manually](#)" on page D-2 for details.
-

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](#)
- [Disabling Removable Storage Service on Windows Media Servers](#)
- [Extracting Oracle Secure Backup from OTN Download on Windows](#)
- [Running the Oracle Secure Backup Windows Installer](#)
- [Configuring Oracle Secure Backup](#)
- [Configuring Firewalls for Oracle Secure Backup on Windows](#)
- [Upgrade Installation on Windows 32-Bit](#)
- [Upgrade Installation on Windows x64](#)
- [Uninstalling Oracle Secure Backup on Windows](#)

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 (Transmission Control Protocol/Internet Protocol)**.
- 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 **Small Computer System Interface (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 that are to be used in the media server role, follow the procedures 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 (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/technology/products/secure-backup>

4. Click **Free Download**.

The Oracle Technology Network Developer License Terms page appears.

5. Read **Export Controls on the Programs** and select the **Yes, I accept...** option.

Read the **Oracle Technology Network Development License Agreement** and click **I Accept**.

The Oracle Secure Backup Downloads page appears.

6. Select the **Accept License Agreement** option, and click the link for the version of Oracle Secure Backup release 10.2 specific to your operating system.

Note: If you have more than one operating system in your environment, then you must perform multiple downloads of the Oracle Secure Backup release 10.2 software.

7. The Oracle Secure Backup release 10.2 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 10.2.

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 Real Application Clusters (RAC) environment, then you must install Oracle Secure Backup on each node in the cluster.

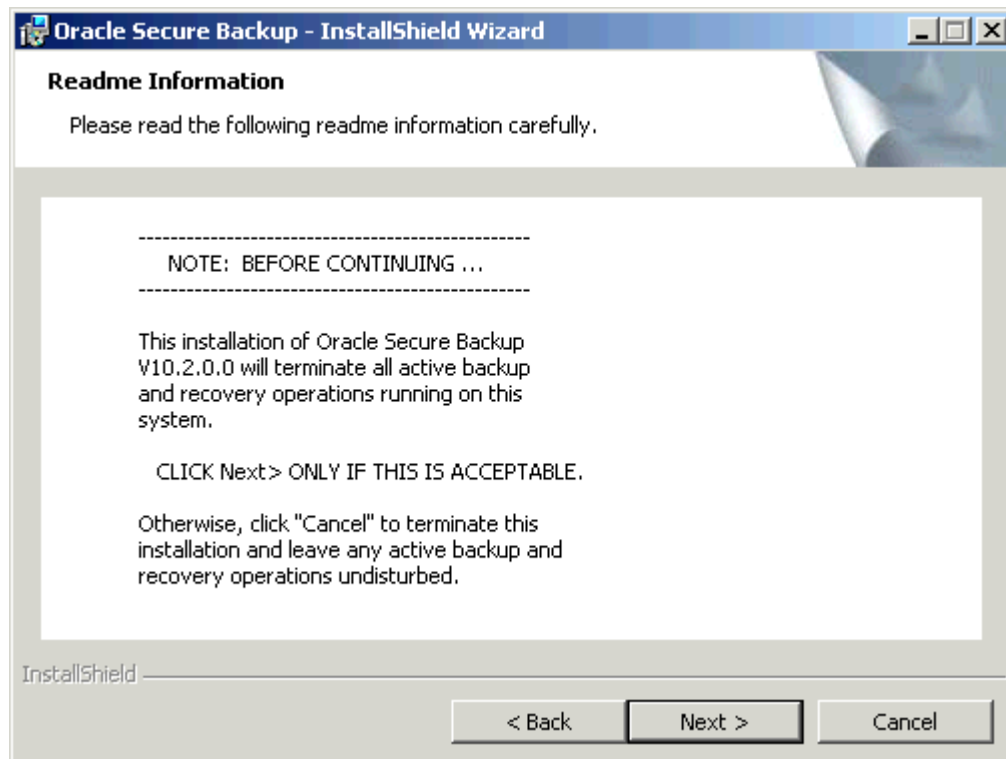
1. Select one of the following install options:
 - If you are installing Oracle Secure Backup from a CD-ROM, then insert the CD-ROM. If AutoPlay is enabled, then the setup.exe program starts automatically and opens the Oracle Secure Backup Setup Wizard.
If Windows AutoPlay is not enabled, then open the 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.



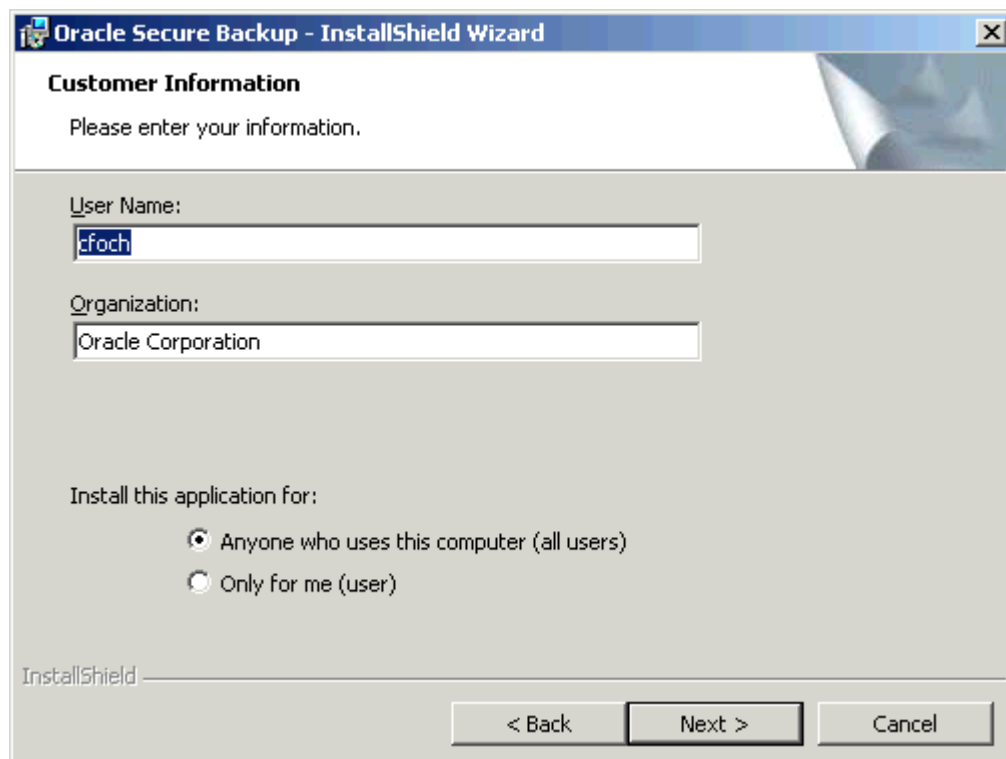
2. Click **Next** to continue.

The Readme Information screen appears.



3. Click **Next** to continue.

The Customer Information screen appears.

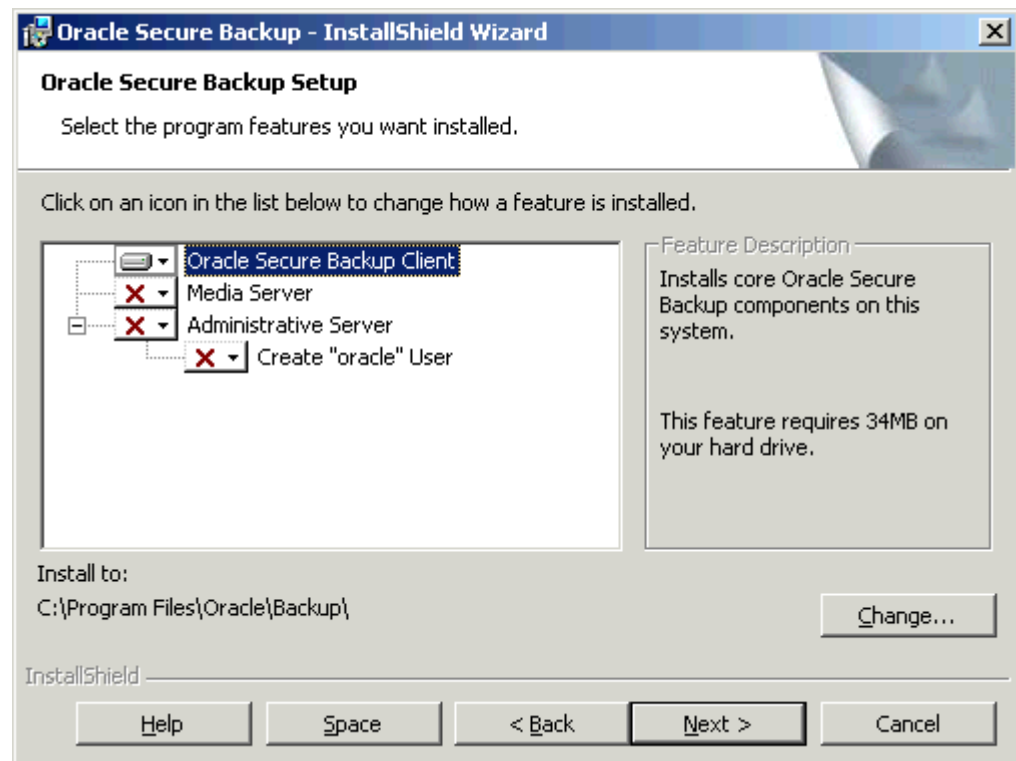


4. Enter your customer information as follows:

- a. Enter your name in the **User Name** field.
- b. Enter the name of your company in the **Organization** field.
- c. Select one of the following:
 - **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.



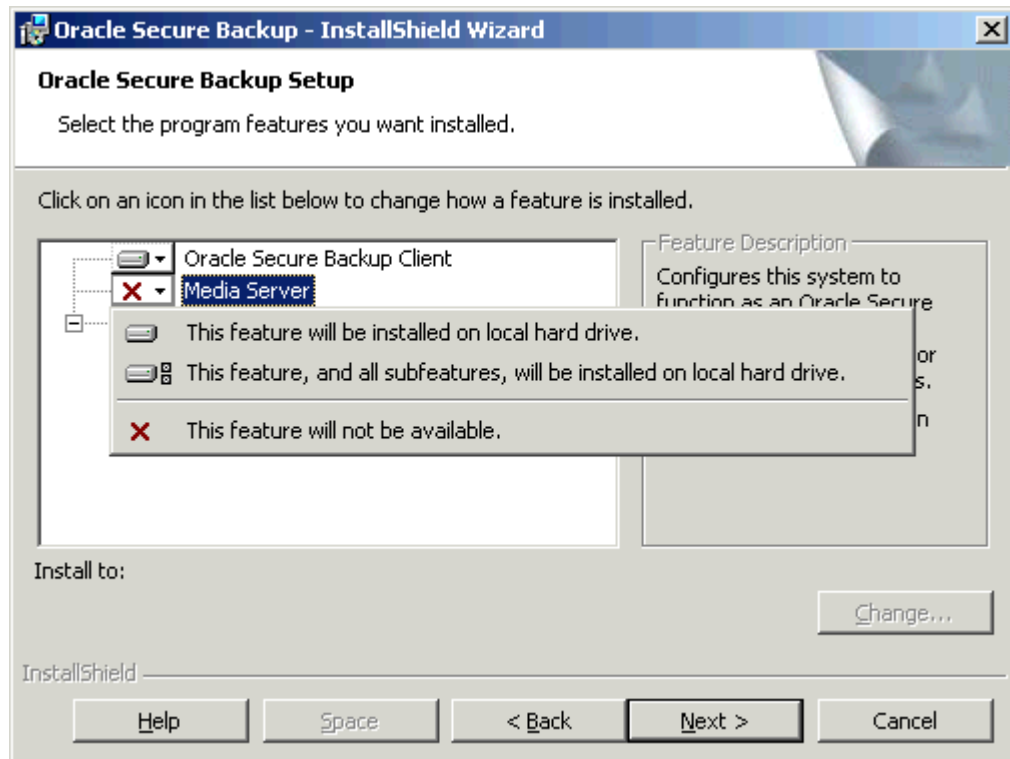
5. A single host can assume multiple **roles**. Roles in Oracle Secure Backup are additive rather than exclusive. You have the following options when choosing roles:

- To install the Windows host as **client** only, click **Next** and go to step 10.

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

- To install the Windows host as a **media server**, proceed to Step 6.
- To install the Windows host as an **administrative server** but not as a media server, skip to Step 7.

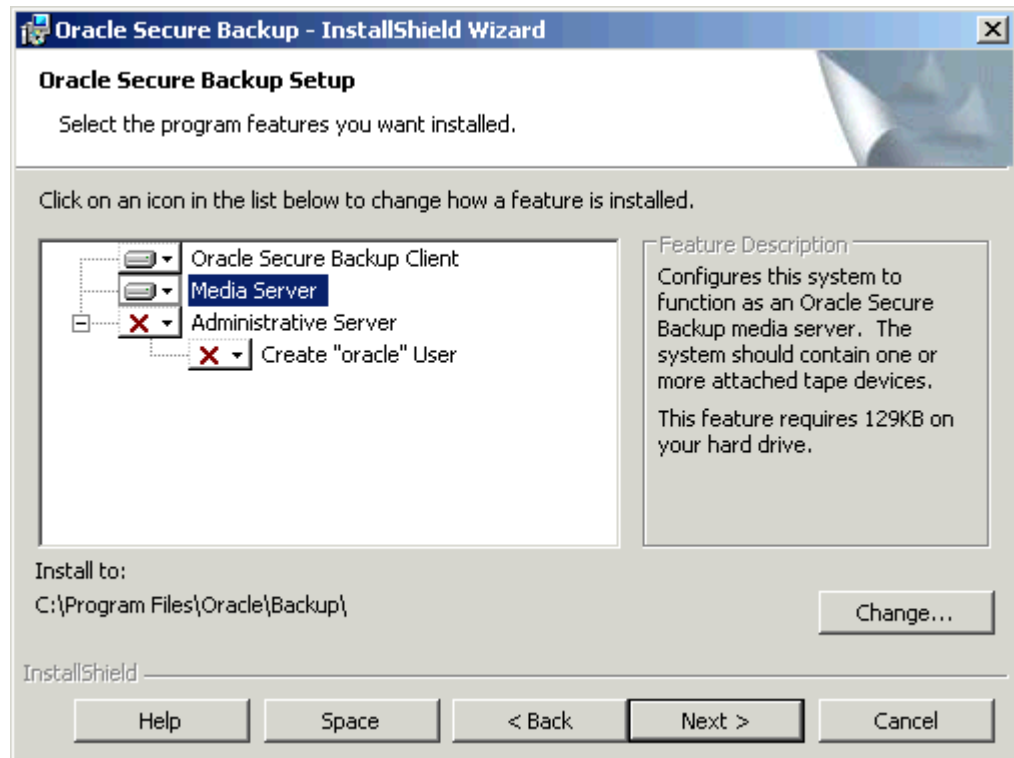
- To install the Windows host as a media server, click the Media Server menu and select **This feature will be installed on local hard drive**.



Selecting this option removes the X from the Media Server icon and includes the media server software in the installation. Installation of the media server software on a host, however, does not make the host a media server. To make a host a media server, the `admin` user must grant the media server role to the host with the Oracle Secure Backup Web tool or the `obtool chhost` command after Oracle Secure Backup is installed.

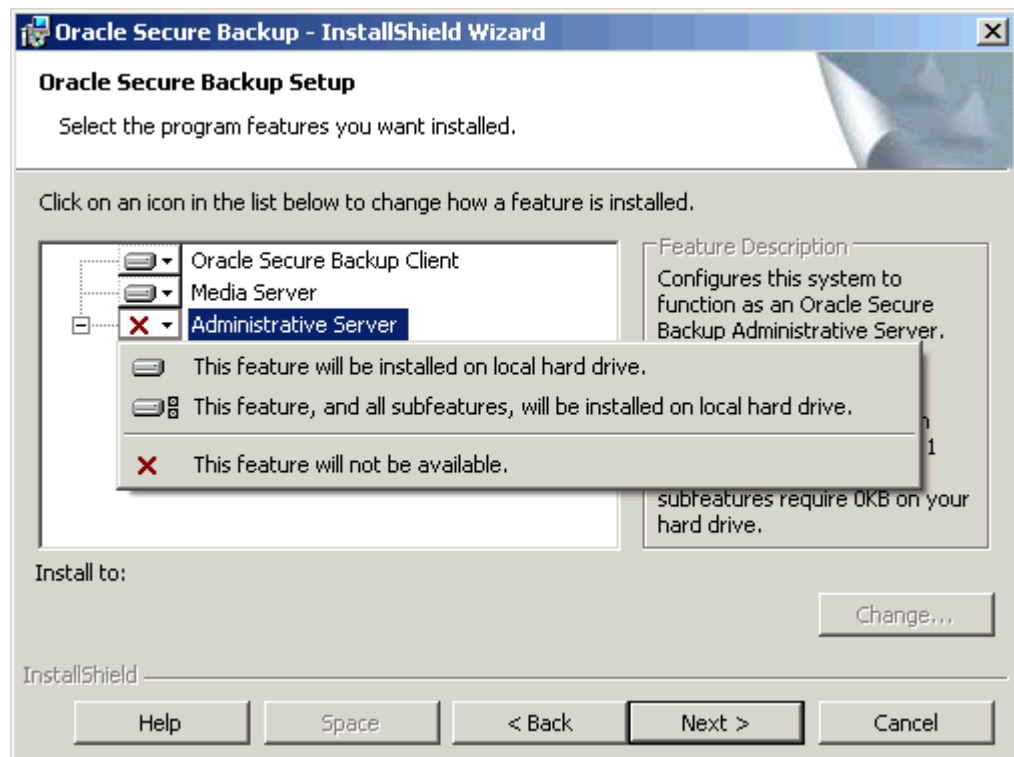
See Also:

- ["Adding the Media Server Role to an Administrative Server"](#) on page 5-6
- ["Adding a Host to the Administrative Domain"](#) on page 5-4
- *Oracle Secure Backup Reference* for complete syntax and semantics for the `obtool chhost` command

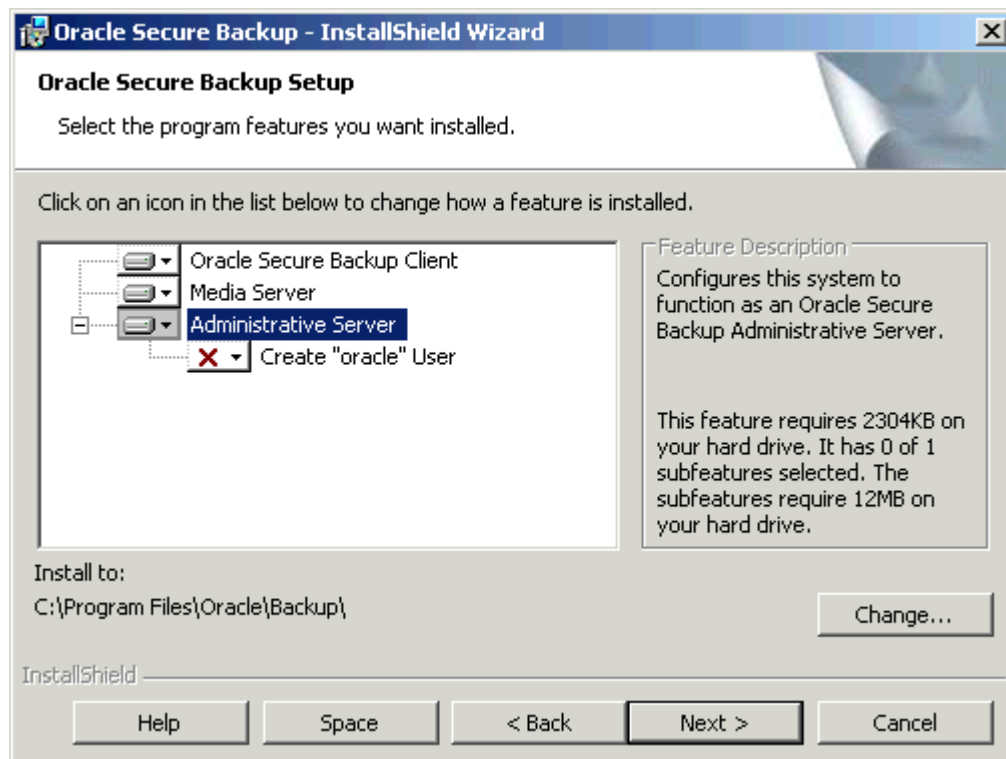


To configure the Windows host to also be an administrative server, proceed to Step 7. Otherwise, click **Next** and go to step 10.

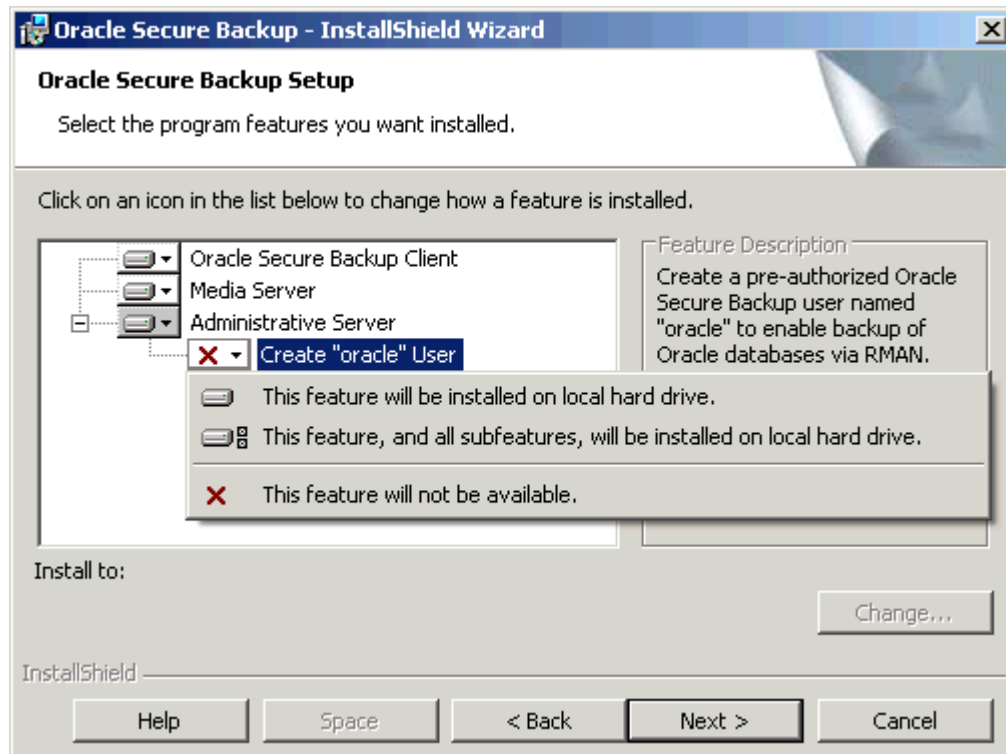
7. To install the Windows host as an administrative server, click the administrative server menu and select **This feature will be installed on local hard drive**.



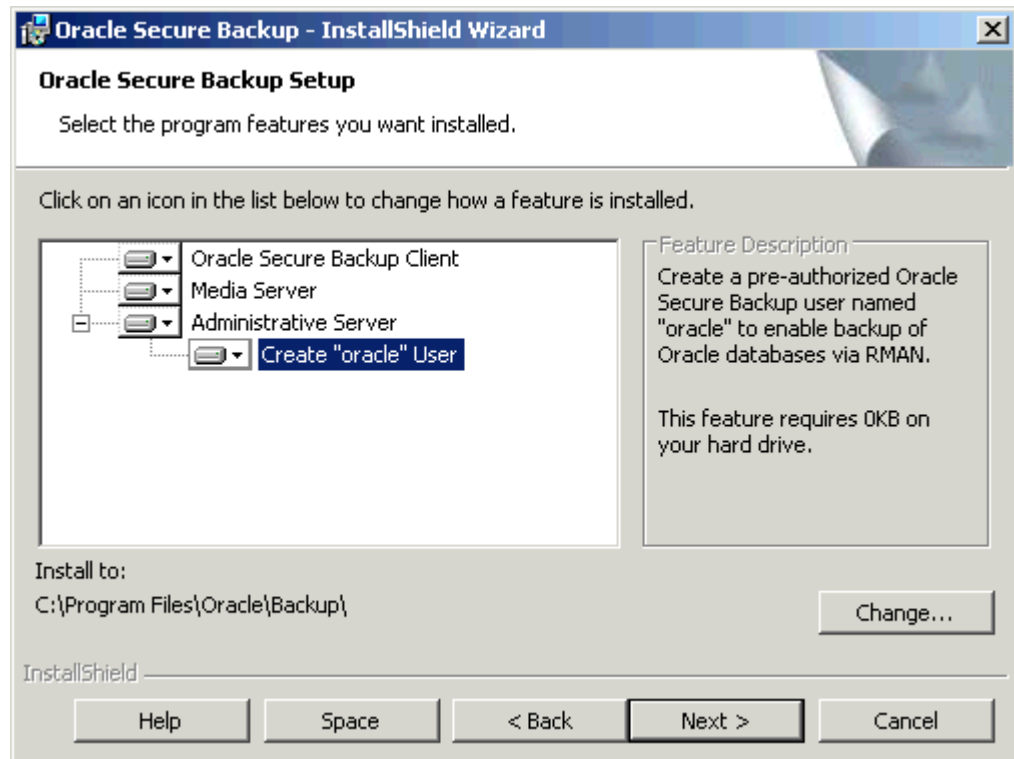
Selecting this option removes the X from the administrative server icon and includes the administrative server in the installation.



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, username and password) that are valid at the client with required privileges after you complete the Oracle Secure Backup installation. Otherwise, Oracle Secure Backup is unable to 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 into 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 steps 6 and 7, 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.

The screenshot shows a Windows-style dialog box titled "Oracle Secure Backup - InstallShield Wizard". The main heading is "Encryption Key Store Password". Below the heading, it says "Please enter a password for the Oracle Secure Backup encryption key store." There are two text input fields. The first is labeled "Password for encryption key store:" and contains ten asterisks. The second is labeled "Re-type password for verification:" and also contains ten asterisks. Below the fields, a hint is provided: "Hint: Oracle suggests you choose a password of at least 8 characters in length, containing a mixture of alphabetic and numeric characters. The password cannot be null, and the maximum password length is 16 characters." At the bottom left, the "InstallShield" logo is visible. At the bottom right, there are three buttons: "< Back", "Next >", and "Cancel".

8. Enter a password for the Oracle Secure Backup encryption key store in the **Password for encryption key store** 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.

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

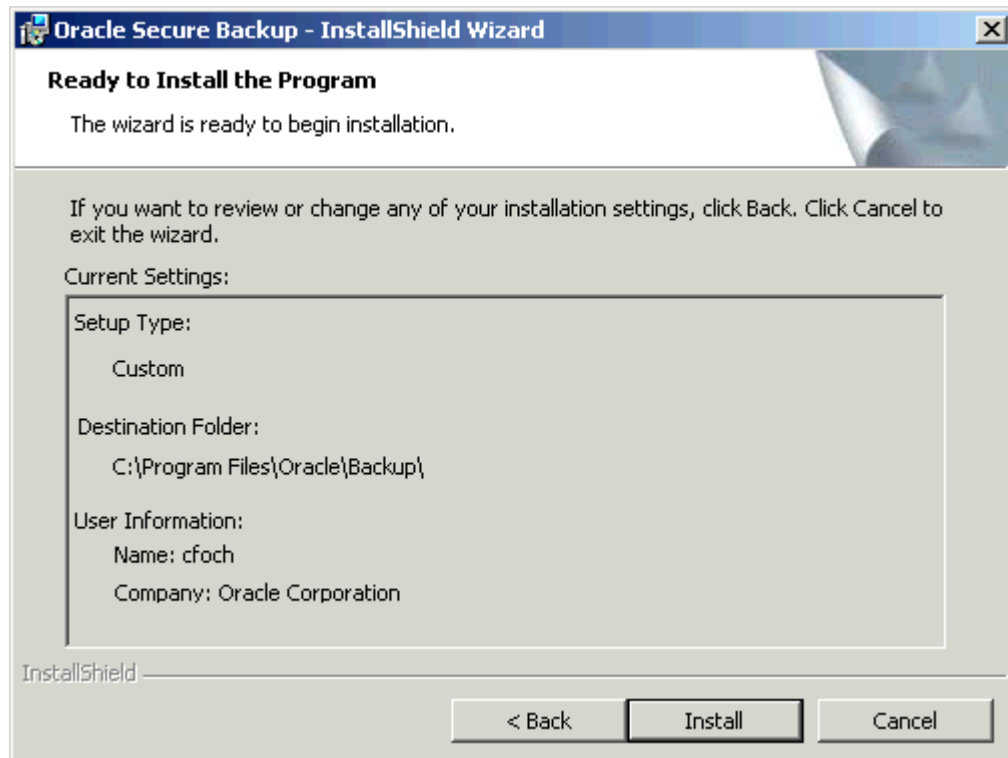
Enter the password again in the **Re-type password for verification** field.

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.

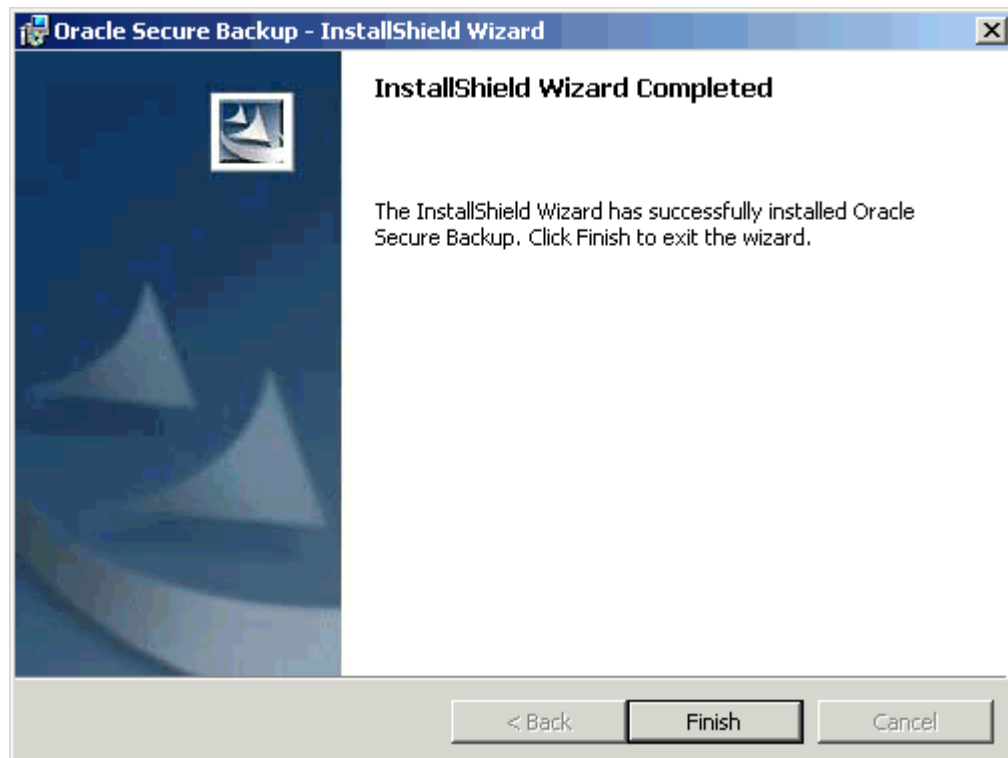
Click **Next**.

The Ready to Install the Program screen appears.



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

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



11. 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 3-13.

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:

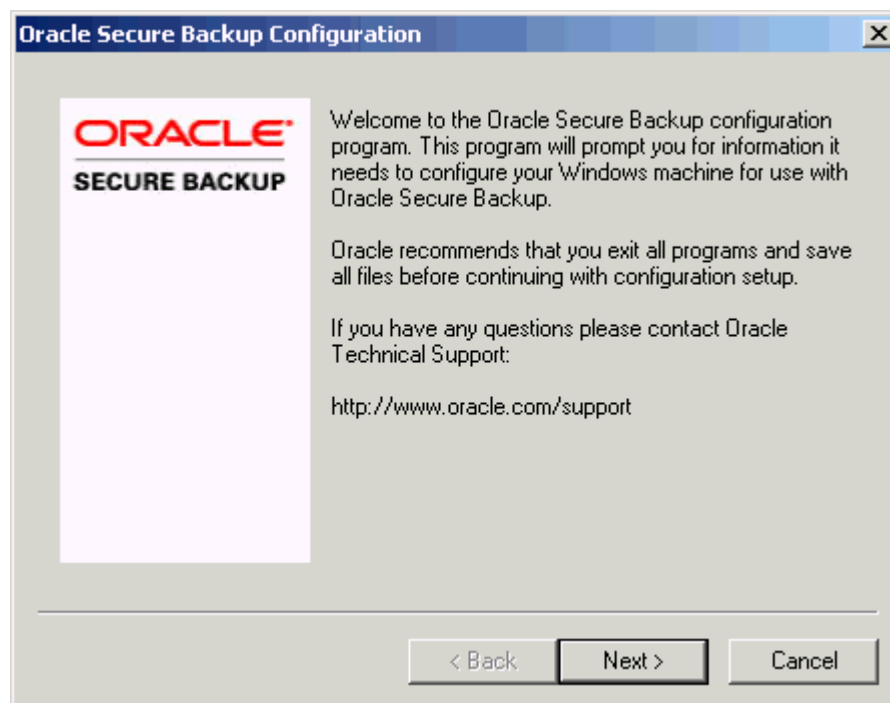
- Select **Start > All Programs > Oracle Secure Backup > Oracle Secure Backup Configuration**
- Enter `obcfg` at the command line

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

1. Start the Oracle Secure Backup Configuration utility.

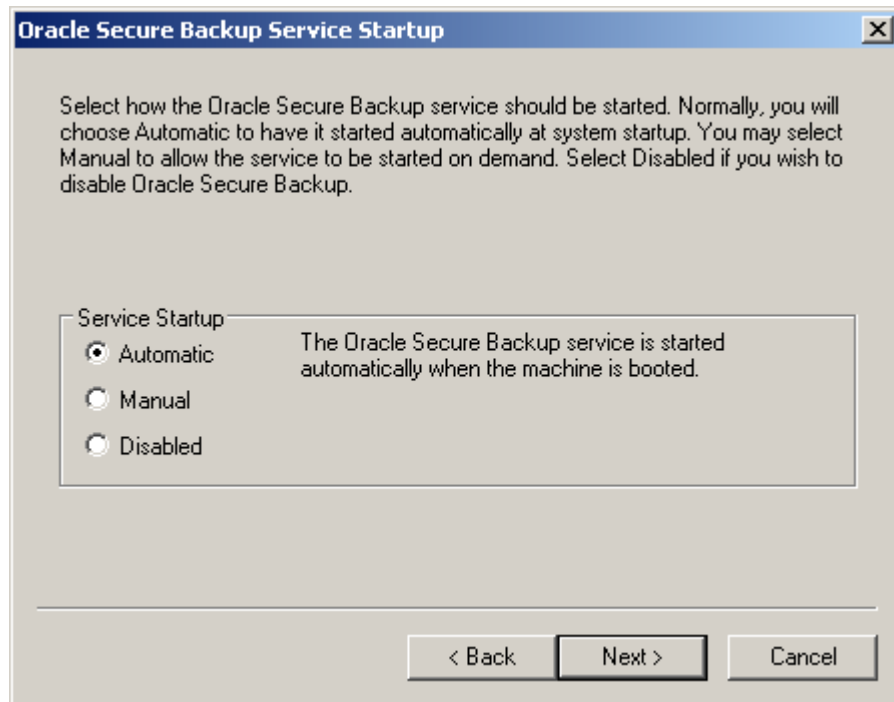
Note: This step is unnecessary if you are configuring Oracle Secure Backup on a Windows host for the first time, because the Oracle Secure Backup Configuration utility starts automatically at the end of the Oracle Secure Backup software installation process.

The Oracle Secure Backup Configuration welcome screen appears.

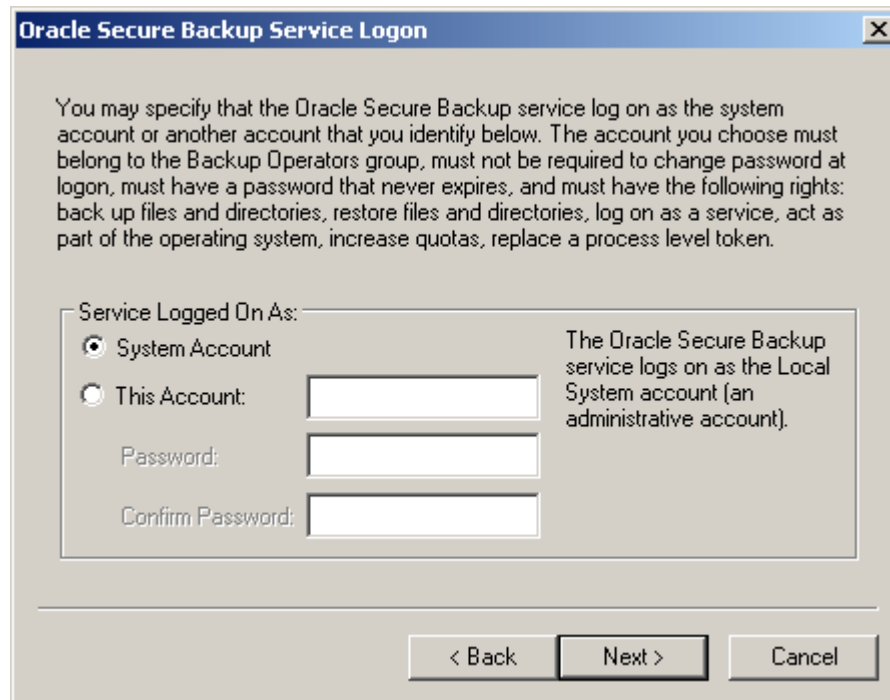


2. Click **Next**.

The Oracle Secure Backup Service Startup screen appears.



3. Select one of the following 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 the following options:

- **System Account**

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

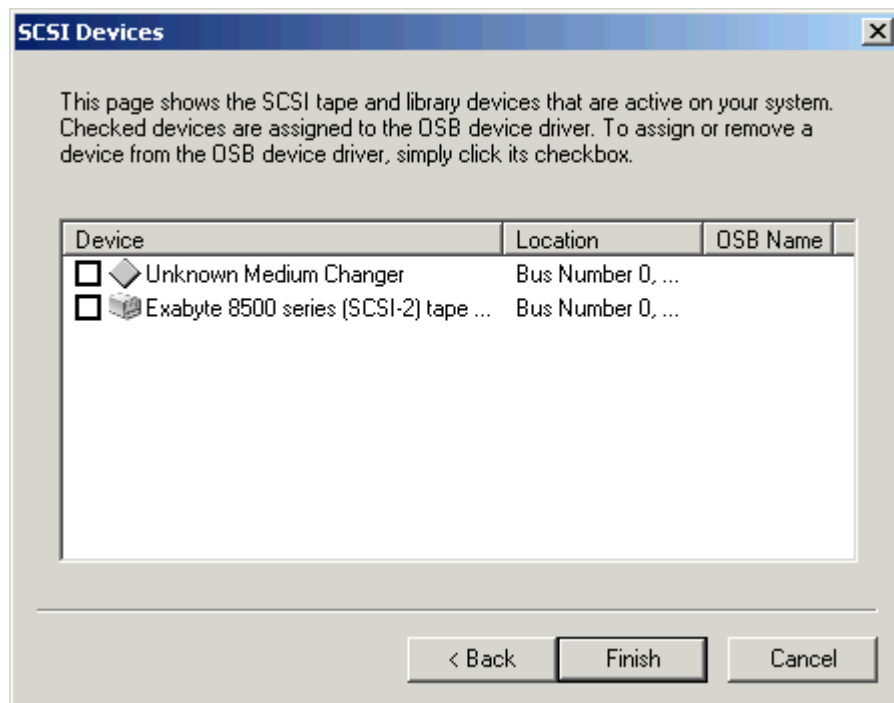
- **This Account**

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.

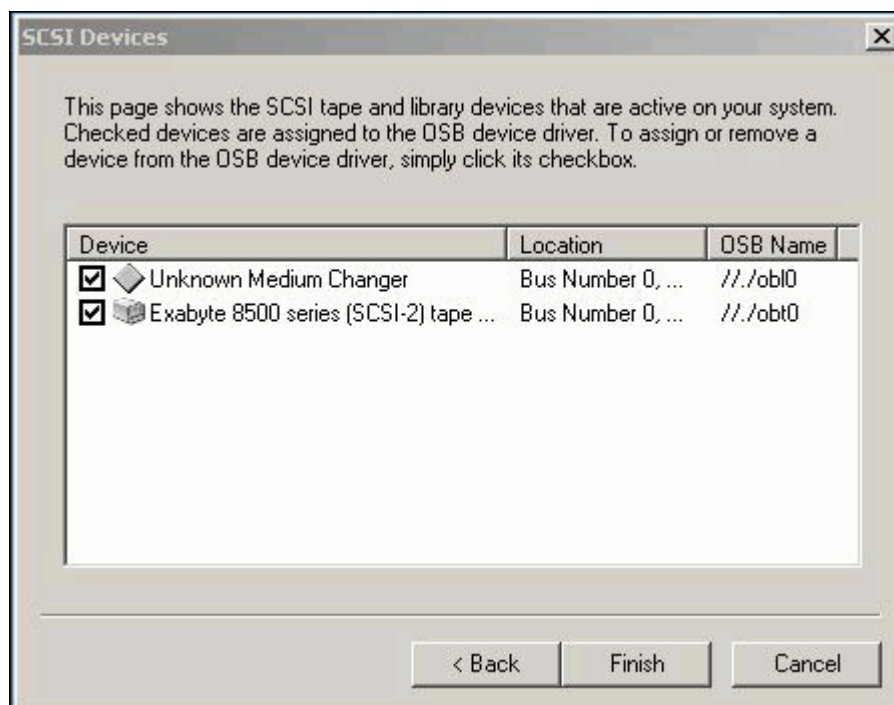
If you choose this option, then you must ensure that the Windows user account you select meets the following criteria:

- 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 drive** and **tape library** devices that you want assigned 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 will need 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 ports 400 and 10000, as well as 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.

Upgrade Installation on Windows 32-Bit

You can upgrade your Windows 32-bit administrative server, media servers, and clients from Oracle Secure Backup release 10.2.0.1 to Oracle Secure Backup release 10.2.0.2 simply by running the Oracle Secure Backup release 10.2.0.2 installer. This is called an *upgrade installation*. The installer detects the existing installation of Oracle Secure Backup and runs the uninstaller for the previous version automatically before beginning the new installation.

When you upgrade your administrative server from Oracle Secure Backup release 10.2.0.1 to Oracle Secure Backup release 10.2.0.2, the uninstaller will display the following prompt:

```
This system was configured as an Oracle Secure Backup Administrative Server.
```

```
Oracle Secure Backup creates files specific to this administrative
domain in the "admin" directory. Would you like to keep these files
in case you reinstall Oracle Secure Backup?
```

```
If you choose "Delete" all files related to Oracle Secure Backup
will be removed from this system. If you choose "Keep" the files
specific to this administrative domain will be retained.
```

You *must* choose the **Keep** option for the admin directory files. Selecting the **Delete** option will cause the installation to be incomplete, and then you must uninstall and

reinstall Oracle Secure Backup to complete the installation. If you do not want to save the existing admin directory files, then you must exit the installation, uninstall Oracle Secure Backup release 10.2.0.1, and select the **Delete** option. After you have uninstalled Oracle Secure Backup release 10.2.0.1, you can install Oracle Secure Backup release 10.2.0.2 by running the Oracle Secure Backup release 10.2.0.2 installer.

You can upgrade your Windows 32-bit administrative server and clients from Oracle Secure Backup release 10.1 to Oracle Secure Backup 10.2.0.2 with an upgrade installation, so long as the administrative server is not also a media server.

When you upgrade your administrative server from Oracle Secure Backup release 10.1 to Oracle Secure Backup release 10.2.0.2, you should select the **Keep** option to keep the existing configuration of your administrative domain. If you select the **Delete** option, then the Oracle Secure Backup release 10.2.0.2 installation will succeed, but you will have to re-enter all of your Oracle Secure Backup configuration settings.

An upgrade installation will not successfully upgrade a media server or an administrative server that is also a media server from Oracle Secure Backup release 10.1 to Oracle Secure Backup release 10.2.0.2. In this special case, you can use the following procedure:

1. Uninstall the existing Oracle Secure Backup software.

See Also: ["Uninstalling Oracle Secure Backup on Windows"](#) on page 3-19

2. Select the **Keep** option if you are upgrading an administrative server that is also a media server.
3. Restart the host.
4. Run the Oracle Secure Backup release 10.2.0.2 installer.

Upgrade Installation on Windows x64

Different upgrade installation procedures must be used for Windows x64 than those described in ["Upgrade Installation on Windows 32-Bit"](#) on page 3-17. You can use the following procedure to upgrade a Windows x64 administrative server or client, so long as the administrative server is not also a media server:

1. Uninstall the existing Oracle Secure Backup software, selecting the **Keep** option if you are upgrading an administrative server.

See Also:

- ["Uninstalling Oracle Secure Backup on Windows"](#) on page 3-19
- ["Upgrade Installation on Windows 32-Bit"](#) on page 3-17 for more information on the **Keep** option

2. Run the Oracle Secure Backup release 10.2.0.2 installer.

You can use the following procedure to upgrade a Windows x64 media server or administrative server that is also a media server:

1. Uninstall the existing Oracle Secure Backup software, selecting the **Keep** option if you are upgrading an administrative server that is also a media server.
2. Restart the host.
3. Run the Oracle Secure Backup release 10.2.0.2 installer.

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**. Choose one of the following:

- Click **Delete** if you do not want to retain the administrative domain files.
- Click **Keep** if you want 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 if you want to reinstall the Oracle Secure Backup software later.

Oracle Secure Backup is now uninstalled from your host.

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](#)
- [Using the Oracle Secure Backup Web Tool](#)
- [Using obtool](#)

Using Oracle Secure Backup in Enterprise Manager

You can use Oracle Enterprise Manager 10g (10.2) or Oracle Enterprise Manager 11g (11.1) 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.

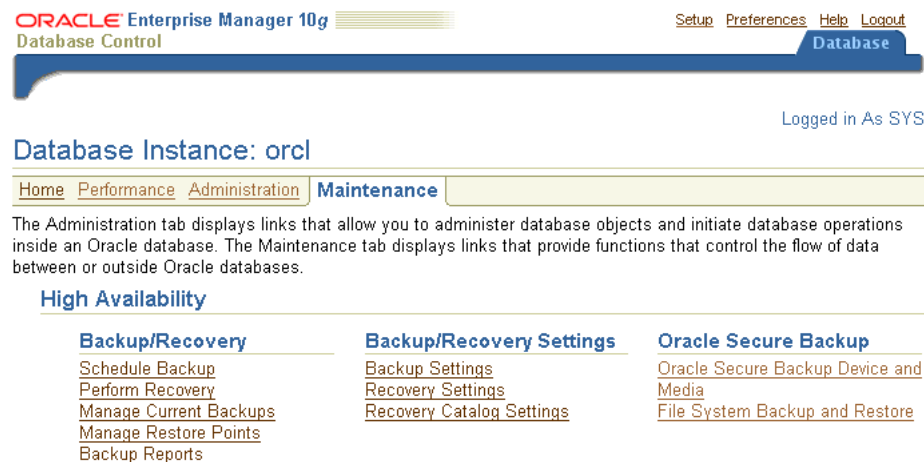
To enable the Oracle Secure Backup section in Oracle Enterprise Manager:

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 running 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 in the domain. This section describes how to get started with the Database Control console.

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 to Oracle Secure Backup that does not require the 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 **Secure Sockets Layer (SSL)**. The **Apache Web server** supplied with Oracle Secure Backup must be running to respond to these requests. Supported web browsers are listed on Certify on Metalink, at the following URL:

<http://metalink.oracle.com/>

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

Page Refreshed Tue Jan 31, 2006, 6:30 pm PST

Failed Jobs 0 jobs in the last 24 hours [Hide failed jobs](#)

ID	Type	Level	Scheduled time	Status
----	------	-------	----------------	--------

Active Jobs 0 jobs in the last 24 hours [Hide active jobs](#)

ID	Type	Level	Scheduled time	Status
----	------	-------	----------------	--------

Pending Jobs 0 jobs in the last 24 hours [Hide pending jobs](#)

ID	Type	Level	Scheduled time	Status
----	------	-------	----------------	--------

Completed Jobs 0 jobs in the last 24 hours [Hide completed jobs](#)

Devices

Type (DTE)	Name	State
library	l2	device not in use
library	ethel mc3	device not in use
drive (ethel_nrst7a:)		device not in use
drive (ethel_nrst8a:)		device not in use
library	u1b1	device not in use
drive (1)	u1t1	device not in use

The main page includes the schedule times and status of recent jobs as well as job IDs, job type, and job level. 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, make sure 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.

- Inactivity logout

This option sets a time period after which the Oracle Secure Backup Web tool logs the user out and returns to the Login page. The default is 30 minutes.

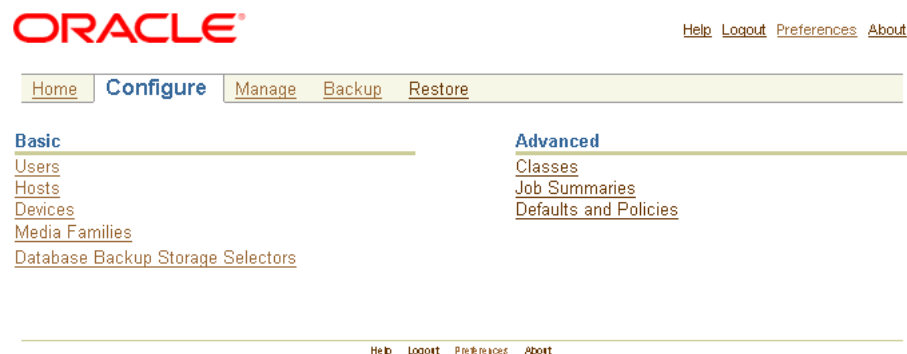
■ About

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 single 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.

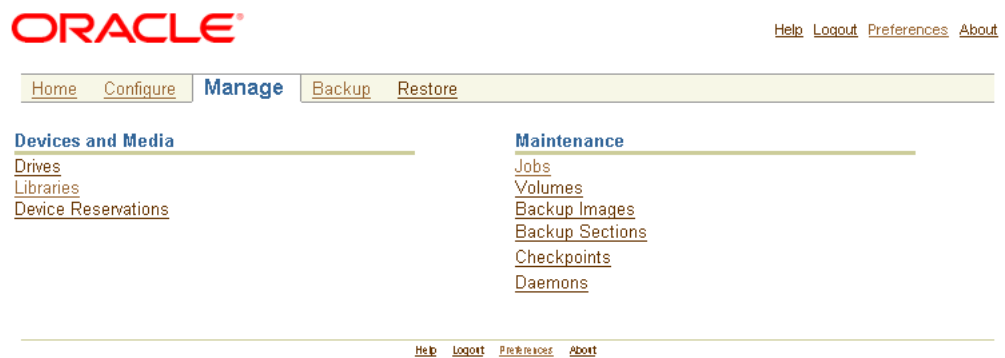
A **job summary** is a generated text file report that tells you whether a backup operation was successful. Oracle Secure Backup can generate and email job summaries detailing the status of each **scheduled backup**.

- **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.

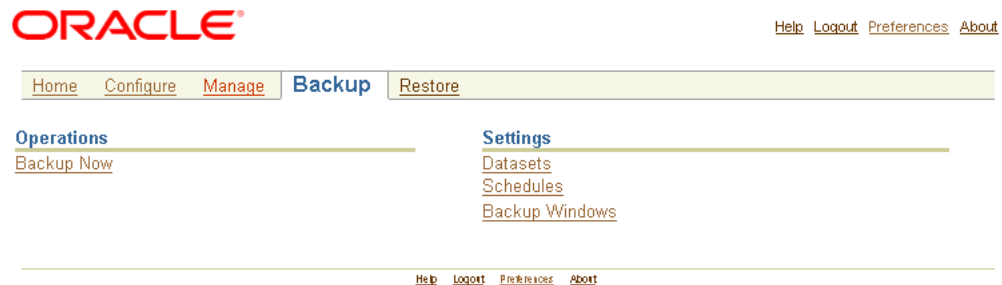
The Maintenance section includes the following links:

- **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 single unit of media, such as 8mm tape. A volume can contain more than one **backup image**.
- **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 that you want to back up.

- **Schedules**

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

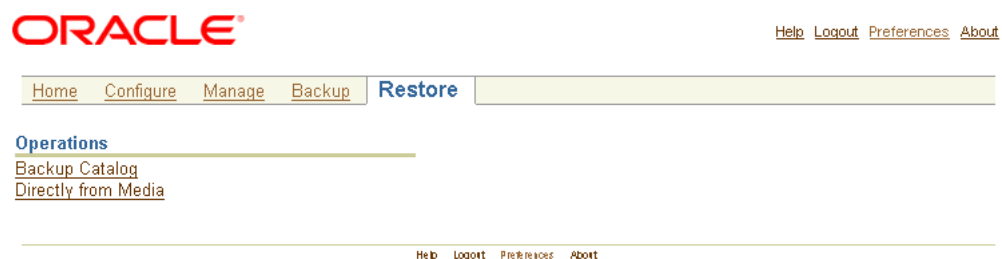
- **Backup Windows**

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 obtool `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 the following 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 10.2.0.0
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 10.2.0.0
login: admin
ob>
```

Starting obtool as a Specific User

You can force **obtool** to use new 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 new 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.

Administrative Domain Configuration and Management

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 of the steps, such as "Adding a Host to the Administrative Domain" on page 5-4, are also useful when managing an existing Oracle Secure Backup administrative domain.

This chapter contains these sections:

- [Administrative Domain Configuration Overview](#)
- [Configuring the Administrative Domain with Hosts](#)
- [Adding Tape Devices to an Administrative Domain](#)

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**. The same tasks can be performed 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, and security options as well as some configuration of 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. This task is described in "Using the Oracle Secure Backup Web Tool" on page 4-4.
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**. This task is described in ["Configuring the Administrative Domain with Hosts"](#) on page 5-2.

Note: If the administrative server is also assigned the media server role, then it is already part of the administrative domain.

- b. Configure the administrative domain to include each **tape device** attached to this host. This task is described in ["Adding Tape Devices to an Administrative Domain"](#) on page 5-11.
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](#)
- [Viewing the Hosts in the Administrative Domain](#)
- [Adding a Host to the Administrative Domain](#)
- [Adding the Media Server Role to an Administrative Server](#)
- [Adding Backup and Restore Environment Variables to an NDMP Host](#)
- [Configuring Preferred Network Interfaces \(PNI\)](#)
- [Pinging a Host](#)
- [Viewing or Editing Host Properties](#)
- [Updating a Host](#)
- [Removing a Host](#)

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 5-9

For hosts that use **NDMP access mode**, such as network-attached storage devices, you have to 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 new 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 that will be used 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 the following:
 - **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 **Access method** field, select one of the following:

- **OB**

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

- **NDMP**

Select this option for devices that support **Network Data Management Protocol (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.

9. 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 16. For hosts such as **Network Attached Storage (NAS)** devices that must use **NDMP** mode, continue to Step 10. Steps 10 through 15 apply only to hosts in NDMP mode.

10. 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

11. 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.
12. In the **Password** list, select one of the following 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.

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.

13. 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.
14. 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.
15. 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.
16. 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.
17. 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 `obtool chhost` command or the Oracle Secure Backup Web tool.

See Also: *Oracle Secure Backup Reference* for complete syntax and semantics for the `obtool 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.

ORACLE® [Help](#) [Logout](#) [Preferences](#) [About](#)

[Home](#) [Configure](#) [Manage](#) [Backup](#) [Restore](#)

[Configure: Hosts](#)

[Add](#) [Edit](#) [Remove](#) [Rename](#) [Update](#) [Ping](#)

Host Name	Status	Roles
osb-test	in service	[admin, client]

☐ Suppress communication with host

[Add](#) [Edit](#) [Remove](#) [Rename](#) [Update](#)

[Classes](#) [Users](#) [Devices](#) [Media Families](#) [Database Backup Storage Selectors](#) [Summaries](#)
[Defaults and Policies](#) [Locations](#) [Rotation Policies](#) [Schedule Location Scan](#) [Volume Duplication Windows](#)
[Volume Duplication Policies](#) [Backup Windows](#)

[Help](#) [Logout](#) [Preferences](#) [About](#)

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

The `Configure: Hosts > host_name` page appears.

ORACLE® [Help](#) [Logout](#) [Preferences](#) [About](#)

[Home](#) [Configure](#) [Manage](#) [Backup](#) [Restore](#)

[Configure: Hosts > osb-test](#)

[Apply](#) [OK](#) [Cancel](#) [Preferred Network Interfaces](#)

Host osb-test

IP interface name(s):

Status:

Roles:

Access method:

Encryption: ☐ required ☒ allowed

Algorithm: ☐ aes128 ☒ aes192 ☐ aes256

Rekey frequency: ☒ duration
☐ never
☐ system default
☐ per backup

Key type: ☒ transparent
☐ use passphrase verify passphrase

TCP/IP buffer size: bytes

Key store: ☐ Add a key to the keystore without making it active

Certificate key size (in bits):

☐ Suppress communication with host

[Apply](#) [OK](#) [Cancel](#) [Preferred Network Interfaces](#)

[Classes](#) [Users](#) [Devices](#) [Media Families](#) [Database Backup Storage Selectors](#) [Summaries](#)
[Defaults and Policies](#) [Locations](#) [Rotation Policies](#) [Schedule Location Scan](#) [Volume Duplication Windows](#)
[Volume Duplication Policies](#) [Backup Windows](#)

[Help](#) [Logout](#) [Preferences](#) [About](#)

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 **Network Data Management Protocol (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 **Preferred Network Interface (PNI)** that identifies the network interface on a

host to use when transmitting backup or restore data to another specified host, or receiving data from that host.

To configure a preferred network interface:

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 will 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 PNI:

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

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 [Network Data Management Protocol \(NDMP\)](#) 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 new 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 **Network Data Management Protocol (NDMP)** mode, such as **Network Attached Storage (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**
- **Preferred Network Interface (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 that you want 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](#)
- [Automatic Discovery of Tape Devices on NDMP Hosts](#)
- [Adding a Tape Device Attachment](#)
- [Multiple Attachments for SAN-Attached Tape Devices](#)
- [Creating Attach Points for Solaris 10 Fibre Channel Devices](#)
- [Pinging a Tape Device](#)
- [Displaying Device Properties](#)
- [Editing Device Properties](#)

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 new **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 **Network Data Management Protocol (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 can be used by the tape device, 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 **obtool** `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 **Small Computer System Interface (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 to hosts. 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 the same as 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 **Network Attached Storage (NAS)** filers. This task is described in "[Automatic Discovery of Tape Devices on NDMP Hosts](#)" on page 5-19.
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 tape 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 and Media section, click **Libraries**.
The Manage: Libraries page appears.

The screenshot shows the Oracle Secure Backup Web tool interface. At the top is the Oracle logo and navigation links: [Help](#), [Logout](#), [Preferences](#), [About](#). Below this is a tabbed interface with tabs: [Home](#), [Configure](#), [Manage](#) (selected), [Backup](#), and [Restore](#). The main content area is titled 'Manage: Libraries'. It features a table with the following data:

Device type (DTE)	Device name
library drive (1)	Exabyte_Library Exabyte_Drive

Below the table is a 'Library commands' section with a dropdown menu currently set to 'Inventory (Library | Drive)'. To the right of the table and commands are buttons: 'Show Properties', 'Error Log', 'List Volumes', and 'Apply'. The footer contains a list of links: [Jobs](#), [Daemons](#), [Device Restrictions](#), [Volumes](#), [Checkpoints](#), [Browse Volumes](#), [Backup Sections](#), [Backup Images](#), [Job Reports](#), [Location Reports](#), [Schedule Volume Duplication](#), [Volumes](#), [Schedule Location Scan](#), and the copyright notice: 'Copyright © 1991, 2007, Oracle. All rights reserved.'

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.
5. Click **Apply**.
A new 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.

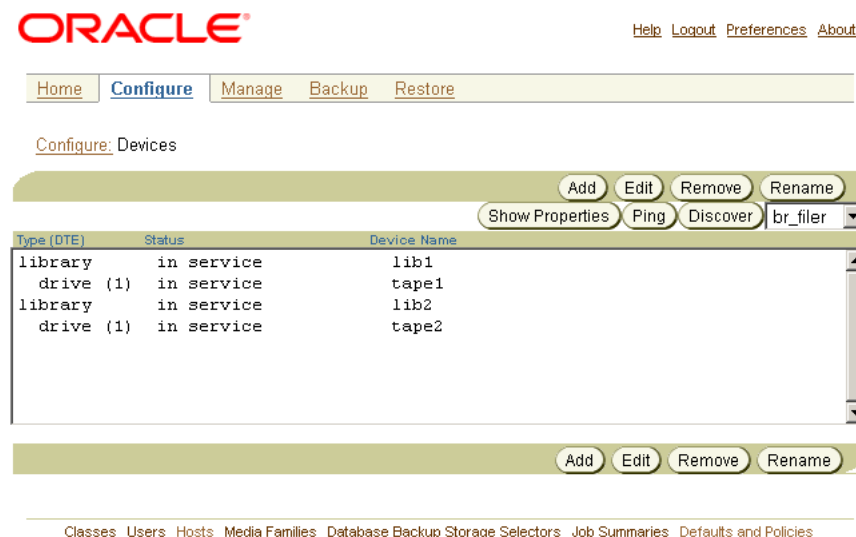
8. Click **OK**.

When the inventory is complete, the Manage: Libraries page reappears and displays a success message. If you want to see the results of the inventory, then 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 **Small Computer System Interface (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 the following 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 as a result of 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 5-11 for more information on World Wide Names

10. In the **Barcode reader** list, select one of the following 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 whether or not 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 will then wait 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 non-zero, and if the number of writable volumes in that tape library has fallen below 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 5-20

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 the following 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 5-20

Configuring a Tape Drive

This section explains how to configure a **tape drive** for use with Oracle Secure Backup. If the tape drive you wish 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 **Small Computer System Interface (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 **Type** list, select **tape**.
7. In the **Status** list, select one of the following 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 as a result of 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 5-11 for more information on World Wide Names
10. If the tape drive is located in a tape library, then select the tape library by name from the **Library** list.
11. In the **DTE** field, enter the **data transfer element (DTE)**.

Note: This option is not available for standalone tape drives.

12. 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.
13. 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**.

14. 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

15. 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.

16. 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.

17. 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.

18. In the **Use list** group, select one of the following options to configure the use list:

- **Storage element range or list**

Select this option for a numerical 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 containing multiple tape drives in which you are using more than one tape drive to perform backups. For example, you might want the tapes in the first half of 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.

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

Automatic Discovery of Tape Devices on NDMP Hosts

Oracle Secure Backup can detect changes in **tape device** configuration for some types of hosts accessed by **Network Data Management Protocol (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 new device with an internally-assigned name and configures a device **attachment** for it.
- If a previously configured tape device has a new 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 new or 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 new or 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.

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 the following that represents the tape device on the host:

- Linux or UNIX **device special file** name

- Windows device name
- **Network Attached Storage (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 device special file or a Windows device file. For example, a **tape library** name might be `/dev/ob10` on Linux and `// . /ob10` on Windows.
4. This step is required only for hosts running certain **Network Data Management Protocol (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 **Small Computer System Interface (SCSI)** device.
 - d. In the **ST controller** field, enter a bus target number.
5. In the **ST lun** field, enter a **SCSI logical unit number (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 **Network Data Management Protocol (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 new 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 **Storage Area Network (SAN)** often has more than one **attachment**, 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 **Network Data Management Protocol (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 **obtool** `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 **Small Computer System Interface (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

Creating Attach Points for Solaris 10 Fibre Channel Devices

Oracle Secure Backup supports the use of Fibre Channel-attached tape devices on media servers running Solaris 10. The Oracle Secure Backup install script and `makedev` utility cannot be used to create the needed links in the `/dev` directory to the device special files for such devices.

The Solaris `sgen` driver must be disabled before installing Oracle Secure Backup. If it is not disabled, then it might attach to your media changer or tape devices and prevent Oracle Secure Backup from attaching to them.

To disable the Solaris `sgen` driver:

1. Comment out all entries in `/kernel/drv/sgen.conf`.

2. Check the `/etc/driver_aliases` file for any `sgen` entries and remove them using the `update_drv` command. Suppose, for example, that `/etc/driver_aliases` contains the following entries:

```
sgen "scsa,08.bfcp"
sgen "scsc,08.bvhci"
```

To remove these entries you would run the following commands:

```
/usr/sbin/update_drv -d -i "scsa,08.bfcp" sgen
/usr/sbin/update_drv -d -i "scsa,08.bvhci" sgen
```

3. Restart your system to release the tape devices.

During installation, the Oracle Secure Backup driver automatically identifies Fibre Channel-attached devices by their World Wide Names. After installing the driver, you must determine the World Wide Name for each device, and then manually create symbolic links in the `/dev` directory that point to the actual attach points for the devices.

To create device special files for Solaris 10 tape devices:

1. Run the Oracle Secure Backup install script on your media server.

When the Oracle Secure Backup driver is installed, the install script prompts:

```
NOTE: The Oracle Secure Backup device driver has been successfully installed.
Would you like to configure (or reconfigure) any Oracle Secure Backup devices
that are attached to dlsun1976 [no]?
```

Enter no.

2. After install completes, run the `dmesg` command and examine the output.

The Oracle Secure Backup driver adds messages to the log that contain the World Wide Names for the Fibre Channel-attached tape devices. The attach points for tape drives contain the string `sgen`. The attach points for tape libraries contain the string `st`.

For example, the following output contains the World Wide Names and corresponding attach points for a tape drive and a tape library:

```
Dec 12 17:12:53 storabck22 scsi: [ID 799468 kern.info] ob30 at fp0: name
w500308c162680e24,1, bus address 6119e8
Dec 12 17:12:53 storabck22 genunix: [ID 936769 kern.info] ob30 is
/pci@1f,4000/fibre-channel@2/fp@0,0/sgen@w500308c162680e24,1
Dec 12 17:12:53 storabck22 scsi: [ID 799468 kern.info] ob31 at fp0: name
w500308c162680e24,0, bus address 6119e8
Dec 12 17:12:53 storabck22 genunix: [ID 936769 kern.info] ob31 is
/pci@1f,4000/fibre-channel@2/fp@0,0/st@w500308c162680e24,0
```

3. For each device listed in the `dmesg` output, assign an **Oracle Secure Backup logical unit number**.
4. Create symbolic links in the `/dev` directory that reference the attach points.

The name for the symbolic link should be `/dev/obtn` for tape drives and `/dev/obln` for tape libraries, where *n* is the Oracle Secure Backup logical unit number you assigned to the device in step 3.

For example, assume that you assign the devices listed in the output of step 2 the Oracle Secure Backup logical unit number 0. The resulting device names are `/dev/obl0` for the tape library, and `/dev/obt0` for the tape drive. The following commands create the required symbolic links in `/dev`:

```
ln -s /devices/pci@1f,4000/fibre-channel@2/fp@0,0/sgen@w500308c162680e24,1:fp0
/dev/obl0
ln -s /devices/pci@1f,4000/fibre-channel@2/fp@0,0/st@w500308c162680e24,0:fp0
/dev/obt0
```

5. Use the `mkdev` command in `obtool` or the Oracle Secure Backup Web tool to add these devices to your administrative domain.

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 can be used by Oracle Secure Backup; if it is not in service, then it cannot be used by Oracle Secure Backup. 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.

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, as well as 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](#)
- [Planning Security for an Administrative Domain](#)
- [Trusted Hosts](#)
- [Host Authentication and Communication](#)
- [Encryption of Data in Transit](#)
- [Default Security Configuration](#)
- [Configuring Security for the Administrative Domain](#)
- [Managing Certificates with obcm](#)

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 has the ability to 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.

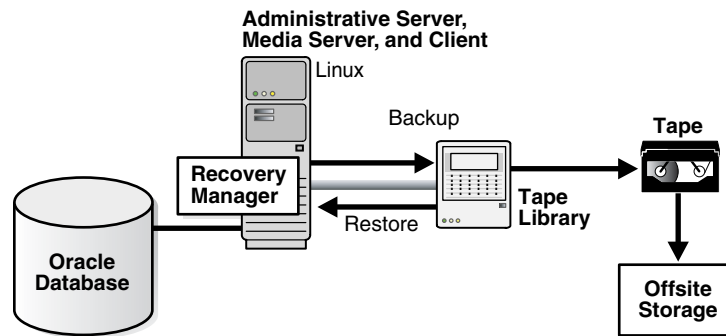
- Isolation of communication medium

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

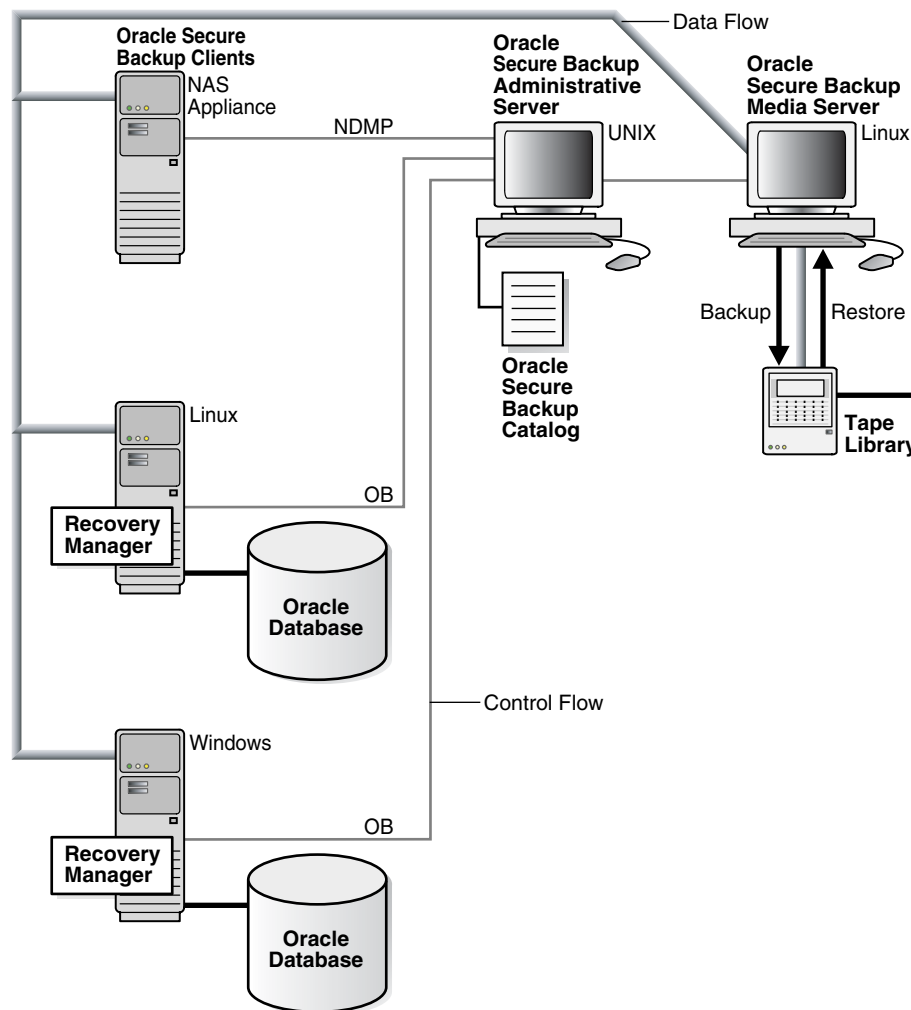
This type of environment is small and isolated from the wider network. The data in this network type is probably on the low end of the sensitivity range. For example, the domain might consist of a server used to host personal Web sites within a corporate network.

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 already 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, more than one **administrative domain**, 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 as well as 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 of the threat 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 new 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 new 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 new host. Instead, you must carry a copy of the signed identity certificate on physical media to the new host and then use the `obcm` utility to import the certificate into the wallet of the new 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 new 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

In Oracle Secure Backup release 10.2 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](#)
- [Authenticated SSL Connections](#)
- [Certification Authority](#)
- [Oracle Wallet](#)
- [Web Server Authentication](#)
- [Revoking a Host Identity Certificate](#)

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**.

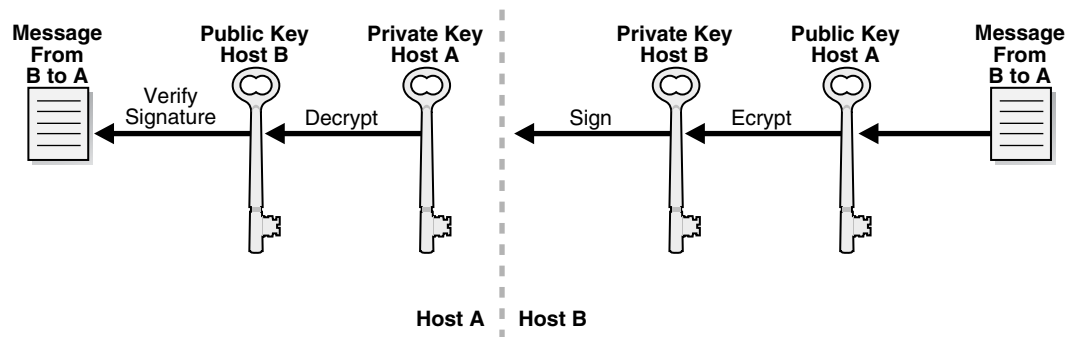
Any host in the domain can use a public key to send an encrypted message to another host. But only the host with the corresponding private key can decrypt the message. A host can use its private key to attach a digital signature to the message. The host

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 with the exception of 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** (observed) 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 as well as 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 new 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 new 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 new 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 new 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 new host.
3. Use obcm on the new 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.

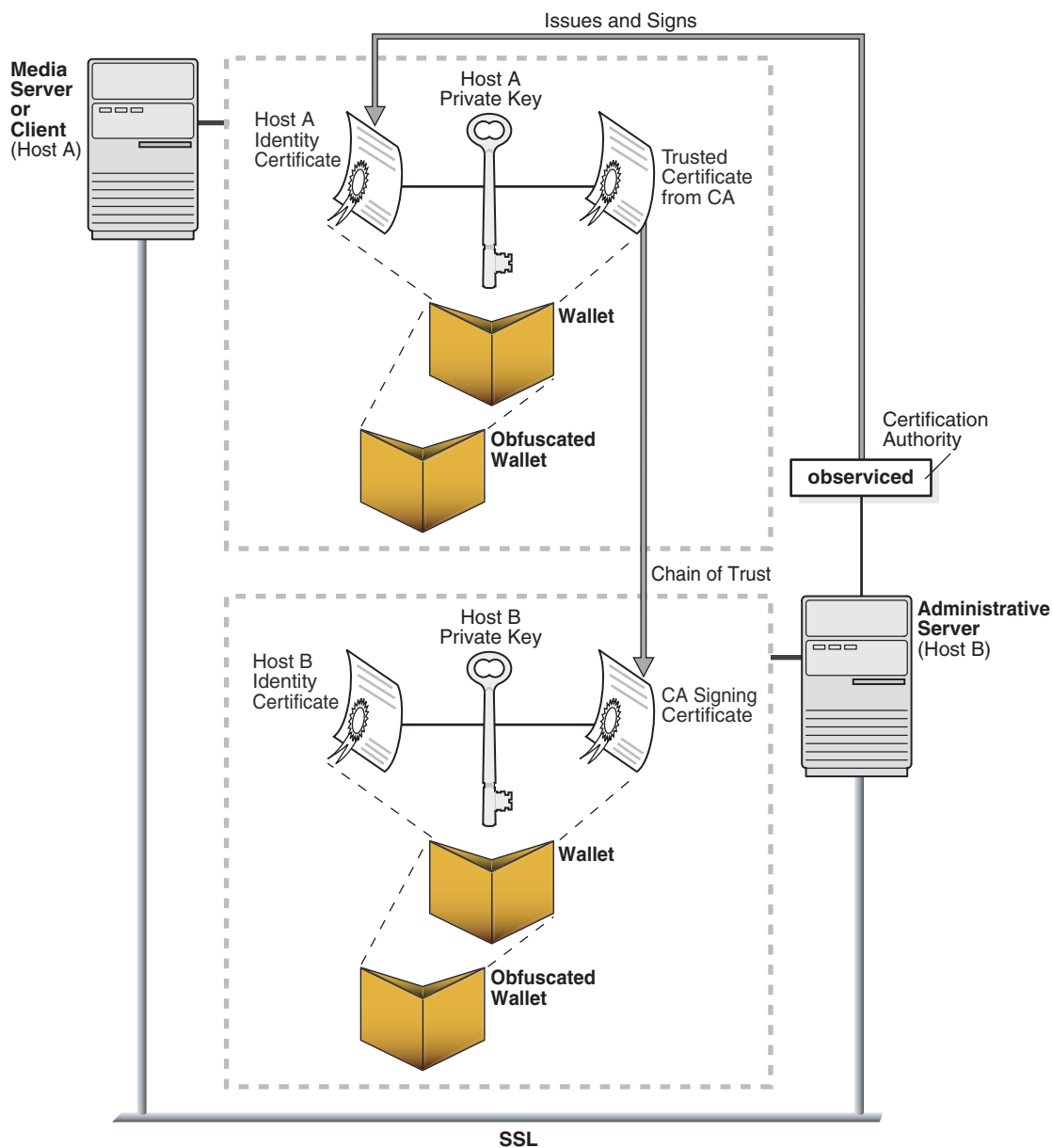
Besides maintaining its password-protected wallet, each host in the domain maintains an **obfuscated wallet**. This version of the wallet does not require a password. The obfuscated wallet, which is scrambled but not encrypted, enables the Oracle Secure Backup software to run without requiring a password during system startup.

Note: To reduce risk of unauthorized access to obfuscated wallets, Oracle Secure Backup does not back them up. The obfuscated version of a wallet is named cwallet.sso. 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
```

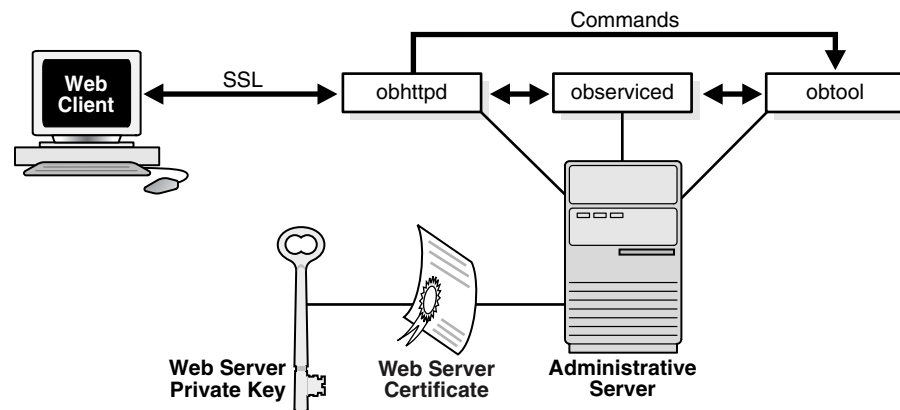
See Also: *Oracle Secure Backup Administrator's Guide* for more information on dataset statements and catalog recovery

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** will 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](#)
- [Setting the Size for Public and Private Keys](#)
- [Enabling and Disabling SSL for Host Authentication and Communication](#)

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 needs to sign the identity certificates for other hosts, and its identity certificate, which it needs 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 new 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 new 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 new 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 new 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 new hosts in **manual certificate provisioning mode**, then you must perform the following steps for each new 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 filename 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 new host.
5. Log on to the host whose wallet will contain 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 filename 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 new host corresponds to the **private key** stored in the wallet with the certificate request. If the keys match, then the new 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 new 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 new 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 new 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 new 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 new host. The following example sets the key size to 4096 bits when configuring new [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 new 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 new 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 filename 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 new host to import a signed **certificate** into the host's **wallet**.

To import a signed **identity certificate** into the wallet of a new host:

1. Log on to the host whose wallet will contain 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 filename 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
```

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/ <i>host_name</i> /	Backup catalog for <i>host_name</i>
admin/log/	Generated log files
admin/log/device/	Log files for devices
admin/log/device/ <i>device_name</i> /	Log files for <i>device_name</i>
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/ <i>device_name</i> /	State for <i>device_name</i>
admin/state/family/	Media family state
admin/state/family/ <i>media_family_name</i>	State for <i>media_family_name</i>
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 <i>host_name</i>
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: <ul style="list-style-type: none"> ■ 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 observed 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 <i>operating_system</i> , where <i>operating_system</i> is a derivative of the operating system name. For example, the directory for Sun Solaris is .bin.solaris.
.drv.operating_system/	Device drivers for <i>operating_system</i>
etc/	Architecture-independent executables for daemons and maintenance tools
.etc.operating_system/	Daemons and utility programs for <i>operating_system</i>
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 <i>operating_system</i> , where <i>operating_system</i> 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 observed 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: <ul style="list-style-type: none"> ■ 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 observed 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 <i>operating_system</i> , where <i>operating_system</i> is a derivative of the operating system name. For example, the directory for Sun Solaris is .bin.solaris.
.drv.operating_system/	Device drivers for <i>operating_system</i>
etc/	Architecture-independent executables for daemons and maintenance tools
.etc.operating_system/	Daemons and utility programs for <i>operating_system</i>
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 observed 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/	<p>Executables or links to executables</p> <ul style="list-style-type: none"> ■ 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 observed 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. <i>operating_system</i> /	Executables for <i>operating_system</i> , where <i>operating_system</i> 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. <i>operating_system</i> /	Daemons and utility programs for <i>operating_system</i>
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 observed 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.

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](#)
- [start daemons at boot](#)
- [identity certificate key size](#)
- [create preauthorized oracle user](#)
- [default UNIX user](#)
- [default UNIX group](#)
- [linux ob dir and solaris64 ob dir](#)
- [linux db dir and solaris64 db dir](#)
- [linux temp dir and solaris64 temp dir](#)
- [linux links and solaris64 links](#)
- [ask about ob dir](#)
- [default protection](#)
- [run obopenssl](#)

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 you do not want the Oracle Secure Backup daemons to start automatically at start time.
yes (default)	Specifies that you want to the Oracle Secure Backup daemons to 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 or not 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 <code>default UNIX user</code> and <code>default UNIX group</code> 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` will be 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
<code>UNIX_user</code>	Specifies the Linux or UNIX operating system username defined in <code>/etc/passwd</code> to which the Oracle Secure Backup user named <code>oracle</code> will be mapped. By default, the Linux/UNIX user is named <code>oracle</code> .

default UNIX group

After the installation is successfully completed and the **administrative domain** has been initialized, a default group will be 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
<code>UNIX_group</code>	Specifies a Linux or UNIX group defined in <code>/etc/group</code> . By default, the Linux/UNIX group is <code>dba</code> .

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.*/observed
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)	Indicates that you want to create the certificate.
no	Indicates that you do not want to create the certificate.

Determining Linux and UNIX SCSI Parameters

For the Linux and UNIX platforms, if you do not know the [Small Computer System Interface \(SCSI\)](#) 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 and UNIX.

This appendix contains these sections:

- [Determining SCSI Device Parameters on Linux](#)
- [Determining SCSI Device Parameters on Solaris](#)

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
```

[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 `scsin`. 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 logical unit number (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 *storabck05 Device Summary*

Device	Host Bus Adapter	SCSI bus address	Target ID	SCSI LUN
Library	0	0	2	0
Tape drive	0	0	4	0

Determining SCSI Device Parameters on Solaris

To determine the **Small Computer System Interface (SCSI)** device parameter information on Solaris, there are two major tasks required:

- To identify the SCSI target ID and **SCSI logical unit number (SCSI LUN)** for each **tape device**, you must probe the SCSI bus of your system for attached tape devices using commands at the console in the Open Boot PROM. This task is described in ["Probing SCSI Target ID and LUN for Media Devices From Solaris Open Boot PROM"](#) on page C-2.

Note: Accessing the Open Boot PROM requires shutting down and restarting the operating system.

- To determine the SCSI bus name-instance for each tape device, you must install the Oracle Secure Backup Solaris device driver, and then view the devices that were recognized by the driver and associate them with the tape device information gathered in the first step.

Note: Installing the Oracle Secure Backup device driver on Solaris is performed after loading Oracle Secure Backup with setup, as described in ["Loading Oracle Secure Backup Software on Linux or UNIX Using setup Script"](#) on page 2-5, but prior to installing Oracle Secure Backup as described in ["Installing Oracle Secure Backup on Linux or UNIX with installob"](#) on page 2-8.

This task is described in ["Viewing SCSI Bus Name-Instance Parameter Values in Solaris"](#) on page C-4.

Probing SCSI Target ID and LUN for Media Devices From Solaris Open Boot PROM

To view **Small Computer System Interface (SCSI)** target ID and **SCSI logical unit number (SCSI LUN)** parameters for a **tape device** from the Solaris Open Boot PROM:

1. Log into the [media server](#) as root.

2. Bring the host to run level 0. For example:

```
# init 0
```

The system shuts down and eventually the Open Boot PROM ok prompt is displayed on the console.

3. At the ok prompt, set the Open Boot auto-boot? variable to false. For example:

```
ok setenv auto-boot? false
auto-boot?      =      false
ok
```

4. At the ok prompt, run the Open Boot reset-all command. For example:

```
ok reset-all
```

The system resets and eventually returns to an ok prompt again.

5. At the ok prompt, run the Open Boot probe-scsi-all command to display the SCSI parameters for all tape devices attached to this host. For example:

```
ok probe-scsi-all
```

Find the information in the output that corresponds to your SCSI devices. For example, this excerpt from the output for dlsun1976 includes the following information for the [tape library](#) and [tape drive](#):

```
/pci@1f,4000/scsi@3,1
Target 0
  Unit 0      Removable Tape      EXABYTE  EXB-85058SQANXR1
Target 1
  Unit 0      Removable Device type 8      EXABYTE  EXB-10e      1.8
```

The output can be interpreted as follows:

- The device tree path for the SCSI bus to which both tape devices are attached is /pci@1f,4000/scsi@3,1. Make a note of this value.

Note: This value is not used directly in Oracle Secure Backup tape device configuration, but is needed when determining the SCSI bus name-instance parameter for each tape device, using the process in ["Viewing SCSI Bus Name-Instance Parameter Values in Solaris"](#) on page C-4.

- The value for Target is the target ID. For example, in this output the target ID of the tape drive is 0, and the target ID of the tape library is 1.
- The value for Unit is the [SCSI logical unit number \(SCSI LUN\)](#). For example, in this output the SCSI LUN of both tape devices is 0.

For this example, assign each tape device the [Oracle Secure Backup logical unit number](#) 0.

Record the discovered parameters and the assigned Oracle Secure Backup logical unit number for each tape device.

6. To restart the host into Solaris, enter the following commands at the ok prompt:

```
ok setenv auto-boot? true
ok reset-all
```

Viewing SCSI Bus Name-Instance Parameter Values in Solaris

To determine the **Small Computer System Interface (SCSI)** bus name-instance parameter to use for each **tape device** in Oracle Secure Backup:

1. Log into your **media server** as `root`.
2. Change directory to the install subdirectory under the **Oracle Secure Backup home**. For example:

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

3. Run the `installdriver` script to install the Oracle Secure Backup driver. For example:

```
# installdriver
NOTE: /usr/local/oracle/backup/.drv.solaris64/ob copied to /usr/kernel/drv/ob
NOTE: /usr/local/oracle/backup/.drv.solaris64/ob.conf copied to
/usr/kernel/drv/ob.conf
NOTE: /usr/local/oracle/backup/.drv.solaris64/ob64 copied to
/usr/kernel/drv/sparcv9/ob
```

NOTE: The Oracle Secure Backup device driver has been successfully installed.

Once installed, the Oracle Secure Backup driver is associated with the tape devices that it can control on this media server.

4. Run the following command to view tape devices associated with the Oracle Secure Backup driver:

```
# du -a /devices|grep ob|cut -f2
/devices/pci@1f,4000/scsi@3,1/ob@0,0:glml
/devices/pci@1f,4000/scsi@3,1/ob@1,0:glml
```

The output contains the needed tape device information.

5. Parse the output from Step 4 using information from the output of probing the SCSI bus in **"Probing SCSI Target ID and LUN for Media Devices From Solaris Open Boot PROM"** on page C-2. For example, consider the line of output that reads:

```
/devices/pci@1f,4000/scsi@3,1/ob@1,0:glml
```

- To identify the SCSI bus used for each tape device in the `du` output, match the device tree paths in the `probe-scsi-all` output to the device tree paths in the output from Step 4 for each tape device.

For example, in this case the bus used for both tape devices is identified in the `probe-scsi-all` output as `/pci@1f,4000/scsi@3,1`, and in the `du` output as `/devices/pci@1f,4000/scsi@3,1`.

- The `ob@` in the path from the `du` output indicates that the tape device is controlled by the Oracle Secure Backup driver.
- The two numbers (in this case, 1, 0) following the `ob@` are the SCSI target ID and **SCSI logical unit number (SCSI LUN)** for each tape device. For this example, the SCSI target ID is 1 and the SCSI LUN is 0. These values correspond to the Exabyte **tape library** on `dlsun1976`, as identified in **"Probing**

[SCSI Target ID and LUN for Media Devices From Solaris Open Boot PROM](#)" on page C-2.

- The value following the colon (:) is the needed SCSI bus name-instance value for this tape device. For this example, the value is `glm1`.

For host `dlsun1976`, parsing both lines of output from Step 4 leads to the final SCSI parameters shown in [Table C-2](#).

Table C-2 *dlsun1976 Tape Device Summary*

Device	Oracle Secure Backup LUN	SCSI Bus Name-Instance	SCSI Target ID	SCSI LUN
Exabyte library	0	glm1	1	0
Exabyte drive	0	glm1	0	0

Manually Configuring Solaris Drivers

This appendix explains how to manually install and uninstall the Oracle Secure Backup kernel device driver for Solaris.

Note: The steps in this chapter are only required for Solaris installations. Oracle Secure Backup automatically uses pass-through drivers for Linux systems.

You must perform the procedures described in this appendix only in one of the following circumstances:

- You chose not to or were unable to use the `installob` installation program described in [Chapter 2, "Installing Oracle Secure Backup on Linux or UNIX"](#).
- You used `installob` but either did not specify a **tape library** or **tape drive**, or you left the `device_list` field blank in your **network description file**. In this case, `installob` did not install the Oracle Secure Backup device driver or create a **device special file**.

Note: It is generally possible to run `installob` again to configure devices for your **media server**. See "[Running installob Again to Create Device Special Files](#)" on page 2-12 for details.

This appendix contains these sections:

- [Installing the Oracle Secure Backup Device Driver Manually](#)
- [Uninstalling the Oracle Secure Backup Device Driver Manually](#)

Installing the Oracle Secure Backup Device Driver Manually

If your **media server** has missing or modified system files, major device numbers already in use, unexpected protection attributes, or other site-specific characteristics, then using `installdriver` to install the driver might not work. Rather, you must install the driver kernel manually.

Perform the following steps to install the Oracle Secure Backup device driver under Solaris 2.8 and later using operating system commands.

Note: These are the same steps performed by the `install/installdriver` shell script. It is strongly recommended that you perform this task by running that script instead of using the manual process described here.

1. Ensure you are logged in as `root`.
2. Check if there is a version of the Oracle Secure Backup driver currently installed:

```
# /usr/sbin/modinfo | grep ob
```

3. If step 2 indicates that the driver named `ob` is installed, then ensure there are no processes using the device driver. If any Oracle Secure Backup **daemons** are running, then stop them at this time using `kill -9`.

4. Uninstall the current driver. For example:

```
# /usr/sbin/rem_drv ob
```

5. Copy the driver from the Oracle Secure Backup Solaris driver directory to `/usr/kernel/drv`. For example:

```
cp /usr/local/oracle/backup/.drv.solaris64/ob /usr/kernel/drv/ob
```

6. Copy the driver's `ob.conf` file:

```
cp /usr/local/oracle/backup/.drv.solaris64/ob.conf /usr/kernel/drv/ob.conf
```

This `ob.conf` file allows Oracle Secure Backup devices to reside at any SCSI target, logical unit number (LUN) 0 or 1, on any bus. You can modify `ob.conf` to specify SCSI targets that correspond only to the devices you want to configure for use by Oracle Secure Backup. Also, you might be required to modify the `ob.conf` file to include LUNs other than 0 or 1 for devices to be claimed by the Oracle Secure Backup driver.

7. Copy the 64-bit version of the driver to `/usr/kernel/drv/sparcv9`. For example:

```
cp /usr/local/oracle/backup/.drv.solaris64/ob64 /usr/kernel/drv/sparcv9/ob
```

8. Add the driver to the system using `add_drv`:

```
/usr/sbin/add_drv -m '* 0666 bin bin' ob
```

9. Use `installob` or `makedev` to create a **device special file** for each **tape library** and **tape drive**.

See Also: ["Creating Device Special Files with makedev"](#) on page 2-12

Uninstalling the Oracle Secure Backup Device Driver Manually

You might be required to uninstall the Oracle Secure Backup driver from your operating system. In most cases, you should use the Oracle Secure Backup `uninstallob` program. See ["Uninstalling Oracle Secure Backup on Linux or UNIX"](#) on page 2-15 for more information.

Note: Because some of the following procedures involve renaming and deleting files, you should make a backup copy of the appropriate directory trees (/etc/conf, /stand/build, and so on) before proceeding.

To uninstall the Oracle Secure Backup driver on Solaris:

1. Log into the host as `root`.
2. Ensure that no **tape drive** or **tape library** is in use.
3. Run the `rem_drv` command to uninstall the Oracle Secure Backup driver. For example:

```
# /usr/sbin/rem_drv ob
```

4. Delete the driver files from `/usr/kernel/drv`. For example:

```
# cd /usr/kernel/drv
# rm ob ob.conf sparcv9/ob
```

Oracle Secure Backup and ACSLS

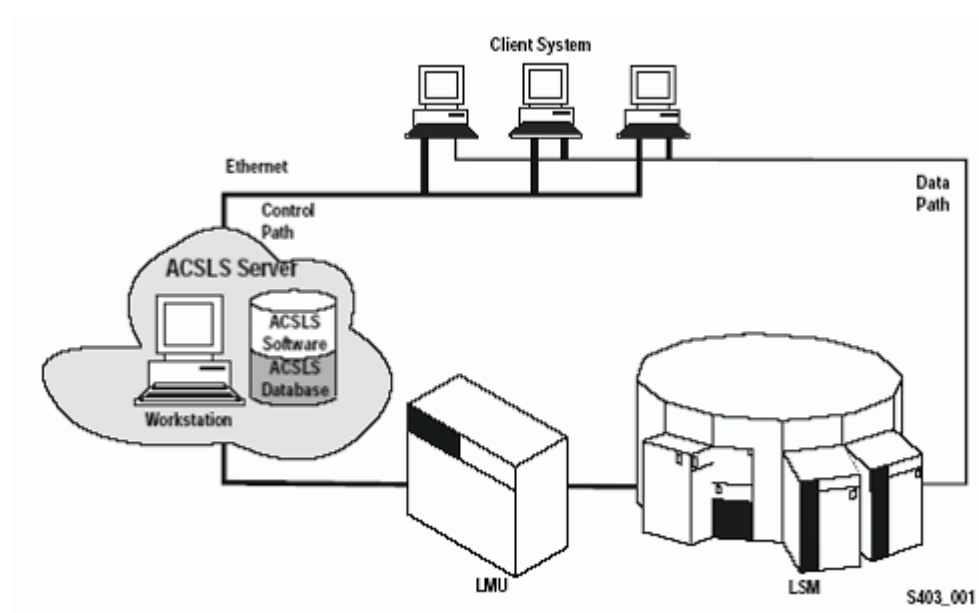
This appendix describes Oracle Secure Backup support for StorageTek Automated Cartridge System Library Software (ACSL). ACSLS is a package of server software that controls one or more Automated Cartridge Systems [tape library](#).

This appendix contains these sections:

- [About ACSLS](#)
- [ACSL and Oracle Secure Backup](#)
- [Communicating with ACSLS](#)
- [Drive Association](#)
- [Volume Loading and Unloading](#)
- [Imports and Exports](#)
- [Access Controls](#)
- [Scratch Pool Management](#)
- [Modified Oracle Secure Backup Commands](#)
- [Unsupported Oracle Secure Backup Commands](#)
- [Installation and Configuration](#)

About ACSLS

[Figure E-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 E-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 E-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 will be 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 `obtool 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. This means that 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** associated with the ACSLS server must also be 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 `hostname/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

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 certificates](#) to new 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 more than one **volume** 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 sections**.

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 by means of 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 specific 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 [certificates](#) maintained by the [Certification Authority \(CA\)](#).

class

A named set of [rights](#) for [Oracle Secure Backup users](#). 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 backups](#), 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 more than one [tape drive](#), 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 is described in a [dataset file](#). 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 together 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 that you want 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 \(NDMP\)](#) as well as configuration changes for such devices.

device special file

A filename in the /dev file system on UNIX or Linux that represents a hardware **tape device**. A device special file 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 as well as permissions and ownership data. An **attachment** consists of a host name and the device special filename 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 in conjunction with 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 or not they have changed since the last backup.

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 certificates](#), 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 levels](#) 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 jobs](#).

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 logical unit number (SCSI LUN)**.

manual certificate provisioning mode

A mode of **certificate** management in which you must manually export the signed **identity certificate** for a new host from the **administrative server**, transfer it to the new host, and manually import the certificate into the **wallet** of the new 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. Note that 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 (Transmission Control Protocol/Internet Protocol)**. 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/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 device special filenames 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 **device special file**. Do not confuse it with **SCSI logical unit number (SCSI LUN)**, which is part of the hardware address of the device.

Oracle Secure Backup user

A defined account 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 network interface that should be used to transmit 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 should be used.

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. Note that 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. Private keys can be used 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 in conjunction 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 (observed) 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 logical unit number (SCSI LUN)

A 3-bit identifier used on a [Small Computer System Interface \(SCSI\)](#) 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 devices](#). 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.

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 single unit of media, such as an 8mm tape. A volume can contain more than one **backup image**.

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 in the case of 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 of time 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

access mode

- about, 1-3
- about NDMP, 1-3
- about primary, 1-3
- selecting, 5-5

ACSLs

- about, E-1
- access controls, E-4
- and obtool, E-2
- cartridges, E-2
- communicating with, E-3
- configuration, E-5
- drive association, E-3
- imports and exports, E-4
- installation, E-5
- modified obtool commands, E-4
- scratch pool, E-3
- scratch pool management, E-4
- unsupported obtool commands, E-5
- volume loading and unloading, E-3

adding

- hosts, 5-4
- hosts in manual certificate provisioning mode, 6-17
- tape device attachments, 5-20

admin user

- creating password during installation on Linux/UNIX, 2-9
- creating password during installation on Windows, 3-11

administrative domain

- configuration outline, 5-1
- configuration overview, 5-1
- defined, 1-2
- host naming, 1-3

administrative server

- about, 1-2
- configuring security, 6-16
- directories, A-1
- files, A-1
- installation on Linux/UNIX, 2-9
- installation on Windows, 3-7
- registering with Oracle Enterprise Manager, 4-3

AIX

installation on, 2-12

uninstallation on, 2-12

Apache Web server

- and network security, 6-13

assets

- identifying for network security, 6-2

attachments

- about, 1-10
- adding for tape devices, 5-20
- displaying device attachment properties, 5-22
- pinging for tape devices, 5-21
- raw device names, 5-21
- setting NDMP version, 5-21

authorization types

- NDMP servers, 5-5

automated certificate provisioning mode

- about, 6-7, 6-11
- and network security, 6-17

automatic discovery

- tape devices, 5-11

automatic tape drive cleaning

- configuring, 5-17

automatic volume ejection, 5-16

automount mode

- about, 1-8
- setting for tape drive, 5-18

B

backup encryption

- enabling, 6-14

backup environment

- and network security, 6-3

backup type

- setting for NDMP hosts, 5-6

barcode readers

- configuring, 5-15

barcodes

- about, 1-8

block size

- about, 1-6
- and restore operations, 1-7

blocking factor

- about, 1-6
- and restore operations, 1-7
- setting for tape drive, 5-19

setting maximum for tape drive, 5-19

C

certificate provisioning

about automated mode, 6-7

about manual mode, 6-7

Certification Authority (CA), 6-10

and network security, 6-9

certkeysize policy, 6-19

client

defined, 1-2

installation on Linux/UNIX, 2-9

installation on Windows, 3-5

client host

directories, A-5

files, A-5

clients

configuring security, 6-17

configuration file parameters

ask about osb dir, B-5

create preauthorized oracle user, B-2

customized obparameters, B-1

default protection, B-5

default UNIX/LINUX group, B-3

default UNIX/LINUX user, B-3

identity certificate key size, B-2

linux db dir, B-4

linux links, B-4

linux ob dir, B-3

linux temp dir, B-4

run obopenssl, B-6

solaris db dir, B-4

solaris links, B-4

solaris ob dir, B-3

solaris temp dir, B-4

solaris64 db dir, B-4

solaris64 links, B-4

solaris64 ob dir, B-3

solaris64 temp dir, B-4

start daemons at boot, B-2

configuring

about tape device names, 5-11

ACSLs, E-5

administrative server security, 6-16

barcode readers, 5-15

client security, 6-17

discovering tape devices on NDMP hosts, 5-20

editing host properties, 5-9

host access mode, 5-5

host key sizes, 5-5

host roles, 5-5

host status, 5-4

hosts, 5-2

key sizes, 5-5

media server security, 6-17

naming tape drives, 5-18

naming tape libraries, 5-15

NDMP authorization type, 5-5

NDMP host backup type, 5-6

NDMP host environment variables, 5-8

NDMP host password type, 5-6

NDMP host port number, 5-6

NDMP protocol version, 5-6

pinging hosts, 5-9

preferred network interfaces, 5-8

removing a host, 5-10

tape device attachments, 5-20

tape devices, 5-11

tape drive automount mode, 5-18

tape drive blocking factor, 5-19

tape drive data transfer element, 5-18

tape drive error rate, 5-18

tape drive maximum blocking factor, 5-19

tape drive status, 5-18

tape drive storage element use list, 5-19

tape drive usage, 5-19

tape drive World Wide Name (WWN), 5-18

tape drives, 5-11, 5-17

tape libraries, 5-11, 5-15

tape library status, 5-15

tape library World Wide Name (WWN), 5-15

testing tape device attachments, 5-21

updating hosts, 5-10

viewing host properties, 5-9

Web tool Hosts page, 5-3

configuring automatic tape drive cleaning, 5-17

D

daemons

automatic start, B-2

listening ports, 3-17

obacslib, E-3, E-4

obacsssid, E-3

obhttpd, 6-13

obrobotd, E-3

observed, 6-10

Web tool Manage page, 4-9

data encryption

about, 6-14

data transfer element

defined, 1-9

tape drive configuration, 5-18

device driver

installing manually on Solaris, D-1

removing on Solaris, D-2

uninstalling on Solaris, D-2

device names

about, 1-10

device special files

creating with makedev, 2-12

devices

about discovering automatically, 5-19

directories

administrative server, A-1

client, A-5

home, A-1

media server, A-4

discovering devices

- about, 5-19
- displaying
 - device attachment properties, 5-22
 - Web tool Backup page, 4-10
 - Web tool Configure page, 4-7
 - Web tool Devices page, 5-14
 - Web tool Home page, 4-5
 - Web tool Hosts page, 5-3
 - Web tool Manage page, 4-8
 - Web tool Restore page, 4-10
- DTE
 - See* data transfer element

E

- editing
 - host properties, 5-9
 - tape device properties, 5-24
- e-mail address
 - adding on Linux/UNIX, 2-10
- encryptdataintransit policy, 6-14, 6-16
- encryption in transit, 6-14
- environment variables
 - setting for NDMP host, 5-8
- error rate
 - setting for tape drive, 5-18
- exporting
 - identity certificates, 6-21

F

- Fibre Channel parameters
 - prerequisites, 2-3
- filers
 - support for SSL, 6-9
- firewalls
 - configuring after installation on Windows, 3-17

H

- home directory
 - location, A-1
- hosts
 - about configuration, 5-2
 - access modes, 1-3
 - adding environment variables for NDMP host, 5-8
 - adding in manual certificate provisioning mode, 6-17
 - adding with Web tool, 5-4
 - configuring access modes, 5-5
 - configuring key sizes, 5-5
 - configuring preferred network interfaces, 5-8
 - configuring roles, 5-5
 - duplicate names, 1-12
 - editing properties, 5-9
 - IP addresses, 5-4
 - naming, 1-3
 - NDMP authorization type, 5-5
 - pinging, 5-9
 - removing, 5-10

- removing preferred network interfaces, 5-9
- setting NDMP backup type, 5-6
- setting NDMP host port number, 5-6
- setting NDMP password type, 5-6
- setting NDMP protocol version, 5-6
- setting status, 5-4
- trusted, 6-8
- updating, 5-10
- viewing properties, 5-9
- Web tool Hosts page, 5-3

- identity certificates
 - distributing, 6-7
 - exporting, 6-21
 - importing, 6-21
 - managing with obcm, 6-21
 - revoking, 6-13
- IEE
 - See* import/export element
- import/export element
 - defined, 1-9
- importing
 - identity certificates, 6-21
- installation
 - overview, 1-13
 - with Oracle Real Application Clusters, 2-2
- installation media
 - about, 1-12
- installation on Linux/UNIX
 - about obparameters, 2-7
 - about oracle user, 2-7
 - admin password, 2-9
 - administrative server, 2-9
 - attach points, 2-14
 - client, 2-9
 - configuring tape devices, 2-10
 - confirming obparameters, 2-8
 - creating OSB home, 2-5
 - disabling SCSI scan software, 2-5
 - downloading software, 2-4
 - e-mail address, 2-10
 - extracting software, 2-4
 - installob, 2-8
 - keystore password, 2-10
 - loading software, 2-5
 - media server, 2-9
 - preparing for, 2-5
 - with Oracle Real Application Clusters, 2-5
- installation on Windows
 - administrative server, 3-7
 - assigning users Windows credentials, 3-9
 - configuring firewalls, 3-17
 - creating oracle user, 3-8
 - creating password for admin user, 3-11
 - creating password for key store, 3-10
 - disabling Removable Storage Service, 3-2
 - disabling SCSI scanning software, 3-1
 - downloading installation software, 3-2

- extracting installation software, 3-2
- media server, 3-6
- preliminary steps, 3-1
- running the installer, 3-3
- selecting host roles, 3-5
- selecting logon account, 3-15
- selecting startup mode, 3-14
- selecting tape devices, 3-16
- with Oracle Real Application Clusters, 3-3
- installation parameters
 - ask about osb dir, B-5
 - create preauthorized oracle user, B-2
 - customized obparameters, B-1
 - default protection, B-5
 - default UNIX/LINUX group, B-3
 - default UNIX/LINUX user, B-3
 - identity certificate key size, B-2
 - linux db dir, B-4
 - linux links, B-4
 - linux ob dir, B-3
 - linux temp dir, B-4
 - run obopenssl, B-6
 - solaris db dir, B-4
 - solaris links, B-4
 - solaris ob dir, B-3
 - solaris temp dir, B-4
 - solaris64 db dir, B-4
 - solaris64 links, B-4
 - solaris64 ob dir, B-3
 - solaris64 temp dir, B-4
 - start daemons at boot, B-2
- installing
 - ACSLs, E-5
 - Solaris device driver, D-1
- installob
 - running, 2-8
 - running multiple times, 2-12
- interfaces
 - about, 1-10
- inventory update, 5-13
- IP addresses
 - configuring a host, 5-4
 - requirements, 1-12

K

- key sizes
 - configuring, 5-5
- key store
 - creating password during installation on Windows, 3-10
- keys
 - setting size, 6-18
- keystore
 - creating password on Linux/UNIX, 2-10

L

- Linux
 - probing SCSI parameters, C-1
- logical unit numbers

- prerequisites, 2-3
- logon account
 - selecting during installation on Windows, 3-15

M

- madev
 - running, 2-12
 - SCSI parameters, 2-13
- malicious users
 - and network security, 6-7
- manual certificate provisioning mode, 6-11
 - about, 6-7
 - adding hosts in, 6-17
 - and network security, 6-17
- manual volume ejection, 5-16
- maximum blocking factor
 - about, 1-6
 - setting for tape drive, 5-19
- media server
 - defined, 1-2
 - directories, A-4
 - files, A-4
 - installation on Linux/UNIX, 2-9
 - installation on Windows, 3-6
- media servers
 - configuring security, 6-17
- medium transport element
 - defined, 1-9
- MTE
 - See* medium transport element
- multiple attachments
 - to storage area networks, 5-22
- multiple data paths, 5-8

N

- names
 - tape devices, 5-11
 - tape drives, 5-11
 - tape libraries, 5-11
- naming
 - tape drives, 5-18
 - tape libraries, 5-15
- NDMP
 - access mode, 5-5
 - supported versions, 1-3
- NDMP access mode
 - about, 1-3
- NDMP authorization type
 - nd5, 5-5
 - negotiated, 5-5
 - text, 5-5
- NDMP hosts
 - adding environment variables, 5-8
 - authorization types, 5-5
 - automatic tape device discovery, 5-11
 - discovering tape devices, 5-20
 - nd5 authorization type, 5-5
 - negotiated authorization type, 5-5
 - setting backup type, 5-6

- setting password type, 5-6
- setting port number, 5-6
- setting protocol version, 5-6
- setting protocol version for device attachment, 5-21
- support for SSL, 6-9
- testing TCP connection, 5-9
- updating, 5-10
- NDMP protocol
 - setting, 5-6
- NDMP text authorization type, 5-5
- NDMP version
 - setting for tape device attachment, 5-21
- network security
 - Apache Web server, 6-13
 - authenticated SSL connections, 6-10
 - automated certificate provisioning mode, 6-17
 - backup environment, 6-3
 - Certification Authority, 6-9
 - Certification Authority (CA), 6-10
 - certkeysize, 6-19
 - configuring clients, 6-17
 - configuring media servers, 6-17
 - configuring the administrative server, 6-16
 - corporate network example, 6-6
 - data center example, 6-4
 - default configuration, 6-15
 - disabling SSL, 6-20
 - distributing identity certificates, 6-7
 - enabling backup encryption, 6-14
 - encryptdataintransit, 6-14, 6-16
 - exporting signed certificates, 6-21
 - host authentication, 6-2, 6-9
 - host communication, 6-9
 - identifying assets, 6-2
 - identifying principals, 6-2
 - identity certificates, 6-9
 - importing identity certificates, 6-21
 - levels, 6-3
 - malicious users, 6-7
 - manual certificate provisioning mode, 6-17
 - obcm utility, 6-21
 - obfuscated wallet, 6-11
 - Oracle wallet, 6-11
 - Oracle wallet passwords, 6-11
 - overview, 6-1
 - planning, 6-2
 - public key cryptography, 6-9
 - revoking an identity certificate, 6-13
 - Secure Sockets Layer, 6-2
 - securecomms, 6-16, 6-20
 - selecting administrative and media servers, 6-6
 - setting key size, 6-18
 - setting key size in obparameters, 6-19
 - setting key sizes in certkeysize security policy, 6-19
 - single-host example, 6-3
 - trusted certificates, 6-10
 - trusted hosts, 6-8
 - using obcm, 6-11

- X.509 certificates, 6-2
- notification
 - e-mail address, 2-10

O

- obcm utility
 - and network security, 6-11
 - exporting certificates with, 6-21
 - importing certificates with, 6-21
 - in manual certificate provisioning mode, 6-18
 - managing certificates, 6-21
- obfirewallconfig.bat, 3-17
- obfuscated wallet
 - and network security, 6-11
- obparameters
 - about, 2-7
 - ask about osb dir, B-5
 - confirming, 2-8
 - create preauthorized oracle user, B-2
 - customized obparameters, B-1
 - default protection, B-5
 - default UNIX/LINUX group, B-3
 - default UNIX/LINUX user, B-3
 - identity certificate key size, B-2
 - linux db dir, B-4
 - linux links, B-4
 - linux ob dir, B-3
 - linux temp dir, B-4
 - run obopenssl, B-6
 - setting key size, 6-19
 - solaris db dir, B-4
 - solaris links, B-4
 - solaris ob dir, B-3
 - solaris temp dir, B-4
 - solaris64 db dir, B-4
 - solaris64 links, B-4
 - solaris64 ob dir, B-3
 - solaris64 temp dir, B-4
 - start daemons at boot, B-2
- obtool
 - about, 1-10, 4-11
 - displaying help, 4-11
 - ending a session, 4-13
 - modified commands for ACSLS, E-4
 - redirecting input from text files, 4-12, 4-13
 - running commands in interactive mode, 4-12
 - running multiple commands, 4-12
 - starting as specific user, 4-13
 - starting in interactive mode, 4-11
 - starting in noninteractive mode, 4-12
 - unsupported commands for ACSLS, E-5
- on-demand volume ejection, 5-16
- operating systems
 - supported, 1-11
- Oracle Enterprise Manager
 - about, 1-10
 - and Oracle Secure Backup, 4-1
 - enabling OSB links, 4-2
 - link to OSB Web tool, 4-3

- registering administrative server, 4-3
- Oracle Real Application Clusters
 - installing Oracle Secure Backup, 2-2
- oracle user
 - creating during installation on Windows, 3-8
 - obparameter, B-2
- Oracle wallet
 - and network security, 6-11
 - obfuscated, 6-11
 - passwords, 6-11
- OSB home
 - creating on Linux/UNIX, 2-5

P

- passwords
 - creating admin password during installation on Linux/UNIX, 2-9
 - creating admin user password during installation on Windows, 3-11
 - creating keystore password during installation on Windows, 3-10
 - creating keystore password on Linux/UNIX, 2-10
 - Oracle wallet, 6-11
 - setting NDMP host password type, 5-6
- pinging
 - hosts, 5-9
 - tape device attachments, 5-21
 - tape devices, 5-24
- port number
 - setting for NDMP host, 5-6
- preferred network interfaces (PNI)
 - configuring, 5-8
 - removing, 5-9
- prerequisites
 - Fibre Channel parameters, 2-3
 - Linux and UNIX, 2-2
 - SCSI Generic driver, 2-2
 - SCSI parameters, 2-3
 - uncompress utility, 2-2
- primary access mode, 1-3, 5-5
- principals
 - identifying for network security, 6-2
- private keys
 - setting size, 6-18
- Probing SCSI parameters
 - on Linux, C-1
- properties
 - displaying for device attachments, 5-22
 - displaying for tape devices, 5-24
- public key cryptography, 6-9
 - in manual certificate provisioning mode, 6-18
- public keys
 - setting size, 6-18

R

- raw device names
 - in tape device attachments, 5-21
- Removable Storage Service

- disabling during installation on Windows, 3-2
- removing
 - hosts, 5-10
 - preferred network interfaces, 5-9
- requirements
 - disk space, 1-11
 - duplicate host names, 1-12
 - host name resolution, 1-12
 - IP addresses, 1-12
 - SCSI Generic driver, 1-12
 - TCP/IP, 1-12
 - WINS, 1-12
- roles
 - selecting during installation on Windows, 3-5
- roles, host, 5-5

S

- scanning software
 - disabling, 5-15, 5-17
- SCSI
 - disabling scanning software, 2-5
- SCSI Generic driver
 - adding, 2-2
 - requirements, 1-12
- SCSI parameters
 - prerequisites, 2-3
 - required by makedev, 2-13
- SCSI scanning software
 - disabling, 5-15, 5-17
 - disabling during installation on Windows, 3-1
- SE
 - See* storage element
- securecomms policy, 6-16, 6-20
- security
 - Apache Web server, 6-13
 - authenticated SSL connections, 6-10
 - automated certificate provisioning mode, 6-17
 - backup environment, 6-3
 - Certification Authority (CA), 6-10
 - certkeysize, 6-19
 - configuring clients, 6-17
 - configuring media servers, 6-17
 - configuring the administrative server, 6-16
 - corporate network example, 6-6
 - data center example, 6-4
 - default configuration, 6-15
 - disabling SSL, 6-20
 - distributing identity certificates, 6-7
 - enabling backup encryption, 6-14
 - encryptdataintransit, 6-14, 6-16
 - exporting signed certificates, 6-21
 - host authentication, 6-2, 6-9
 - host communication, 6-9
 - identifying assets, 6-2
 - identifying principals, 6-2
 - identity certificates, 6-9
 - importing identity certificates, 6-21
 - levels, 6-3
 - malicious users, 6-7

- manual certificate provisioning mode, 6-17
- obcm utility, 6-21
- obfuscated wallet, 6-11
- Oracle wallet, 6-11
- Oracle wallet passwords, 6-11
- planning, 6-2
- public key cryptography, 6-9
- revoking an identity certificate, 6-13
- Secure Sockets Layer, 6-2
- securecomms, 6-16, 6-20
- selecting administrative and media servers, 6-6
- setting key size, 6-18
- setting key size in obparameters, 6-19
- setting key sizes in certkeysize security policy, 6-19
- single-host example, 6-3
- SSL, 6-9
- trusted certificates, 6-10
- trusted hosts, 6-8
- using obcm utility, 6-11
- X.509 certificates, 6-2
- security, overview, 6-1
- setup script
 - about, 2-5
 - running, 2-5
- SSL
 - authenticated connections, 6-10
 - disabling, 6-20
 - support for NDMP, 6-9
- startup mode
 - selecting during installation on Windows, 3-14
- status
 - checking tape devices, 5-24
 - hosts, 5-4
 - setting for tape drives, 5-18
 - setting for tape libraries, 5-15
- storage devices
 - supported, 1-11
- storage element
 - defined, 1-8
- supported
 - NDMP versions, 1-3
 - operating systems, 1-11
 - tape devices, 1-11
 - web browsers, 1-11
- suppress communication with host, 5-10
- system requirements, 1-11

T

- tape devices
 - about, 1-5
 - about attachments, 1-10
 - about multiple device attachments
 - attachments
 - about multiple attachments, 5-22
 - about names, 1-10, 5-11
 - adding device attachments, 5-20
 - automatic discovery, 5-11
 - configuring, 5-11

- configuring during installation on
 - Linux/UNIX, 2-10
- discovering on NDMP hosts, 5-20
- displaying properties, 5-24
- editing properties, 5-24
- pinging, 5-24
- pinging attachments, 5-21
- selecting during installation on Windows, 3-16
- updating inventory, 5-13
- Web tool Devices page, 5-14
- tape drives
 - about discovering automatically, 5-19
 - about logical unit numbers, 2-3
 - about names, 1-10, 5-11
 - adding device attachments, 5-20
 - automatic cleaning, 5-17
 - automatic discovery, 5-11
 - configuring, 5-11, 5-17
 - configuring during installation on
 - Linux/UNIX, 2-10
 - defined, 1-5
 - disabling SCSI scanning software, 5-17
 - displaying properties, 5-24
 - editing properties, 5-24
 - naming, 5-18
 - selecting during installation on Windows, 3-16
 - setting automount mode, 5-18
 - setting blocking factor, 5-19
 - setting data transfer element, 5-18
 - setting error rate, 5-18
 - setting maximum blocking factor, 5-19
 - setting status, 5-18
 - setting storage element use list, 5-19
 - setting usage, 5-19
 - setting world wide names, 5-18
 - supported, 1-11
 - tape formats, 1-7
 - updating inventory, 5-13
 - Web tool Devices page, 5-14
- tape formats, 1-7
- tape libraries
 - about discovering automatically, 5-19
 - about logical unit numbers, 2-3
 - about names, 1-10, 5-11
 - adding device attachments, 5-20
 - automatic discovery, 5-11
 - automatic drive cleaning, 1-8
 - automatic loading, 1-8
 - automatic tape drive cleaning, 5-17
 - configuring, 5-11, 5-15
 - configuring barcode readers, 5-15
 - configuring during installation on
 - Linux/UNIX, 2-10
 - defined, 1-7
 - disabling SCSI scanning software, 5-15
 - displaying properties, 5-24
 - editing properties, 5-24
 - naming, 5-15
 - selecting during installation on Windows, 3-16
 - setting status, 5-15

- setting world wide names, 5-15
- updating inventory, 5-13
- virtual, 1-9
- Web tool Devices page, 5-14
- tape library elements
 - abbreviations, 1-9
 - data transfer, 1-9
 - import/export, 1-9
 - medium transport, 1-9
 - storage, 1-8
- TCP connection
 - testing, 5-9
- TCP/IP
 - requirements, 1-12
- trusted certificates, 6-10
- trusted hosts
 - about, 6-8

U

- uninstalling
 - device driver on Solaris, D-2
 - Oracle Secure Backup on Linux/UNIX, 2-15
 - Oracle Secure Backup on Windows, 3-19
- uninstallob
 - running, 2-15
- updating
 - hosts, 5-10
 - tape device inventory, 5-13
- upgrade installation
 - about, 1-14
 - on Linux or UNIX, 2-14
 - on Windows 32-bit, 3-17
 - on Windows x64, 3-18
- usage
 - setting tape drive usage, 5-19
- use list
 - configuring for tape drive, 5-19
- users
 - default UNIX/LINUX user obparameter, B-3

V

- viewing
 - host properties, 5-9
- virtual tape libraries
 - backup operations, 1-9
 - defined, 1-9
- volumes
 - automatic ejection, 5-16
 - inventory update, 5-13
 - manual ejection, 5-16
 - on-demand ejection, 5-16

W

- web browsers
 - supported, 1-11
- Web tool
 - about, 1-10, 4-4
 - adding a host, 5-4
 - Backup page, 4-10
 - Configure page, 4-7
 - Devices page, 5-14
 - displaying device attachment properties, 5-22
 - displaying tape device properties, 5-24
 - editing host properties, 5-9
 - editing tape device properties, 5-24
 - help, 4-6
 - Home page, 4-5
 - Hosts page, 5-3
 - link to Oracle Enterprise Manager, 4-3
 - logging in, 4-5
 - Manage page, 4-8
 - persistent page links, 4-6
 - pinging tape device attachments, 5-21
 - pinging tape devices, 5-24
 - preferences, 4-6
 - Restore page, 4-10
 - starting, 4-4
 - viewing host properties, 5-9
- Windows installer
 - running, 3-3
- WINS
 - requirements, 1-12
- World Wide Name (WWN)
 - setting for tape drives, 5-18
 - setting for tape libraries, 5-15

X

- X.509 certificates, 6-2