

Veritas NetBackup™ Device Configuration Guide

UNIX, Windows, Linux

Release 8.1

VERITAS™

Veritas NetBackup Device Configuration Guide

Last updated: 2018-02-16

Legal Notice

Copyright © 2018 Veritas Technologies LLC. All rights reserved.

Veritas, the Veritas Logo, and NetBackup are trademarks or registered trademarks of Veritas Technologies LLC or its affiliates in the U.S. and other countries. Other names may be trademarks of their respective owners.

This product may contain third party software for which Veritas is required to provide attribution to the third party ("Third Party Programs"). Some of the Third Party Programs are available under open source or free software licenses. The License Agreement accompanying the Software does not alter any rights or obligations you may have under those open source or free software licenses. Refer to the third party legal notices document accompanying this Veritas product or available at:

<https://www.veritas.com/about/legal/license-agreements>

The product described in this document is distributed under licenses restricting its use, copying, distribution, and decompilation/reverse engineering. No part of this document may be reproduced in any form by any means without prior written authorization of Veritas Technologies LLC and its licensors, if any.

THE DOCUMENTATION IS PROVIDED "AS IS" AND ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT SUCH DISCLAIMERS ARE HELD TO BE LEGALLY INVALID. VERITAS TECHNOLOGIES LLC SHALL NOT BE LIABLE FOR INCIDENTAL OR CONSEQUENTIAL DAMAGES IN CONNECTION WITH THE FURNISHING, PERFORMANCE, OR USE OF THIS DOCUMENTATION. THE INFORMATION CONTAINED IN THIS DOCUMENTATION IS SUBJECT TO CHANGE WITHOUT NOTICE.

The Licensed Software and Documentation are deemed to be commercial computer software as defined in FAR 12.212 and subject to restricted rights as defined in FAR Section 52.227-19 "Commercial Computer Software - Restricted Rights" and DFARS 227.7202, et seq. "Commercial Computer Software and Commercial Computer Software Documentation," as applicable, and any successor regulations, whether delivered by Veritas as on premises or hosted services. Any use, modification, reproduction release, performance, display or disclosure of the Licensed Software and Documentation by the U.S. Government shall be solely in accordance with the terms of this Agreement.

Veritas Technologies LLC
500 E Middlefield Road
Mountain View, CA 94043

<http://www.veritas.com>

Technical Support

Technical Support maintains support centers globally. All support services will be delivered in accordance with your support agreement and the then-current enterprise technical support policies. For information about our support offerings and how to contact Technical Support, visit our website:

<https://www.veritas.com/support>

You can manage your Veritas account information at the following URL:

<https://my.veritas.com>

If you have questions regarding an existing support agreement, please email the support agreement administration team for your region as follows:

Worldwide (except Japan)

CustomerCare@veritas.com

Japan

CustomerCare_Japan@veritas.com

Documentation

The latest documentation is available on the Veritas website:

<https://sort.veritas.com/documents>

Documentation feedback

Your feedback is important to us. Suggest improvements or report errors or omissions to the documentation. Include the document title, document version, chapter title, and section title of the text on which you are reporting. Send feedback to:

NB.docs@veritas.com

You can also see documentation information or ask a question on the Veritas community site:

<http://www.veritas.com/community/>

Veritas Services and Operations Readiness Tools (SORT)

Veritas Services and Operations Readiness Tools (SORT) is a website that provides information and tools to automate and simplify certain time-consuming administrative tasks. Depending on the product, SORT helps you prepare for installations and upgrades, identify risks in your datacenters, and improve operational efficiency. To see what services and tools SORT provides for your product, see the data sheet:

https://sort.veritas.com/data/support/SORT_Data_Sheet.pdf

Contents

Chapter 1	Introducing device configuration	8
	Using this guide	8
	General device configuration sequence	9
	Configuration cautions	9
	About the NetBackup compatibility lists	10
Section 1	Operating systems	11
Chapter 2	AIX	12
	Before you begin configuring NetBackup on AIX	12
	RS/6000 AIX adapter number conventions	13
	About AIX persistent naming support	14
	About configuring robotic control device files in AIX	14
	About device files for SAN Clients on AIX	14
	About configuring tape drive device files in AIX	15
	About choosing a tape driver on AIX	15
	About non-QIC tape drives on AIX	16
	About extended-file marks for drives	16
	About fast-tape positioning (locate-block) on AIX	17
	About no rewind device files on AIX	17
	Creating AIX no rewind device files for tape drives	17
	Using multiple tape densities on AIX	21
	About SPC-2 SCSI reserve on AIX	21
	Disabling SPC-2 SCSI reserve in AIX	22
	About Sony AIT drives on AIX	22
	AIX command summary	23
Chapter 3	HP-UX	25
	Before you begin configuring NetBackup on HP-UX	25
	About robotic control on HP-UX	26
	About HP-UX device addressing schemes	26
	HP-UX tape drive device file requirements for NetBackup	27
	About device drivers and files for HP-UX persistent DSFs	28
	About device drivers for persistent DSFs	29

	About persistent DSFs for robotic control	29
	About persistent DSFs for tape drive access	29
	About persistent DSF pass-through paths	29
	About configuring persistent DSFs	30
	Creating HP-UX persistent DSFs	30
	Upgrading NetBackup to use HP-UX persistent DSFs	31
	Creating persistent DSF pass-through paths	32
	About HP-UX legacy device drivers and files	32
	About HP-UX device drivers for legacy device files	33
	About legacy robotic control device files	33
	About legacy tape drive device files	33
	About legacy pass-through paths for tape drives	34
	Creating device files for SAN Clients on HP-UX	35
	About configuring legacy device files	35
	Creating legacy SCSI and FCP robotic controls on HP-UX	36
	About creating legacy tape drive device files	43
	Creating tape drive pass-through device files	43
	About SPC-2 SCSI reserve on HP-UX	47
	Disabling SPC-2 SCSI reserve in HP-UX	47
	About disabling the HP-UX EMS Tape Device Monitor for a SAN	47
	HP-UX command summary	48
Chapter 4	Linux	49
	Before you begin on Linux	49
	About the required Linux SCSI drivers	50
	About the st driver debug mode	51
	Verifying the Linux drivers	51
	About configuring robot and drive control for Linux	51
	About the Linux robotic control device files	52
	About the Linux tape drive device files	52
	Verifying the device configuration on Linux	52
	About SAN clients on Linux	53
	About SCSI persistent bindings for Linux	54
	About Emulex HBAs	54
	Utilities to test SCSI devices	55
	Linux command summary	55
Chapter 5	Solaris	56
	Before you begin on Solaris	56
	About the NetBackup sg driver	57
	Determining if the NetBackup sg driver is installed	58

	Special configuration for the Oracle StorEdge Network Foundation	
	HBA driver	59
	About binding Fibre Channel HBA drivers	59
	Configuring Solaris 10 x86 for multiple drive paths	60
	Installing/reinstalling the sg and the st drivers	60
	st.conf file example	63
	sg.conf file example	63
	sg.links file example	64
	Configuring 6 GB and larger SAS HBAs in Solaris	65
	Preventing Solaris driver unloading	67
	About Solaris robotic controls	68
	About SCSI and FCP robotic controls on Solaris	68
	Examples of SCSI and FCP robotic control device files on Solaris	
	69
	About Solaris tape drive device files	69
	About Berkeley-style close	71
	About no rewind device files on Solaris	71
	About fast-tape positioning (locate-block) on Solaris	71
	About SPC-2 SCSI reserve on Solaris	72
	Disabling SPC-2 SCSI reserve on Solaris	72
	About nonstandard tape drives	73
	Configuring Solaris SAN clients to recognize FT media servers	73
	Adding the FT device entry to the st.conf file	73
	Modifying the st.conf file so that Solaris discovers devices on two	
	LUNS	74
	Uninstalling the sg driver on Solaris	75
	Solaris command summary	75
Chapter 6	Windows	77
	Before you begin configuring NetBackup on Windows	77
	About tape device drivers on Windows	78
	Attaching devices to a Windows system	78
Section 2	Robotic storage devices	79
Chapter 7	Robot overview	80
	NetBackup robot types	80
	NetBackup robot attributes	81
	ACS robots	81
	TLD robots	82
	Table-driven robotics	83

	Robotic test utilities	84
	Robotic processes	84
	Processes by robot type	85
	Robotic process example	86
Chapter 8	Oracle StorageTek ACSLS robots	88
	About Oracle StorageTek ACSLS robots	89
	Sample ACSLS configurations	89
	Media requests for an ACS robot	93
	About configuring ACS drives	93
	Configuring shared ACS drives	95
	Adding tapes to ACS robots	97
	About removing tapes from ACS robots	97
	Removing tapes using the ACSLS utility	98
	Removing tapes using NetBackup	98
	Robot inventory operations on ACS robots	98
	Configuring a robot inventory filtering on ACS robots	100
	NetBackup robotic control, communication, and logging	101
	NetBackup robotic control, communication, and logging for	
	Windows systems	101
	NetBackup robotic control, communication, and logging for UNIX	
	systems	102
	ACS robotic test utility	106
	acstest on Windows systems	107
	acstest on UNIX systems	107
	Changing your ACS robotic configuration	107
	ACS configurations supported	108
	Multiple ACS robots with one ACS library software host	108
	Multiple ACS robots and ACS library software hosts	109
	Oracle StorageTek ACSLS firewall configuration	110
Chapter 9	Device configuration examples	112
	An ACS robot on a Windows server example	112
	An ACS robot on a UNIX server example	115
Index		119

Introducing device configuration

This chapter includes the following topics:

- [Using this guide](#)
- [General device configuration sequence](#)
- [About the NetBackup compatibility lists](#)

Using this guide

Use this guide to help set up and configure the operating systems of the hosts you use for NetBackup servers. Also use this guide for help with storage devices. This guide provides guidance about NetBackup requirements; it does not replace the vendor documentation.

This guide is organized as follows:

- Information about operating systems.
- Information about robotic storage devices.

Read the "Before you start" sections (if applicable) of the chapters in this guide. These sections provide any important platform-specific instructions or may contain specific instructions or limitations for server types.

Veritas tested the configuration file options in this guide; other configuration settings may also work.

To minimize configuration errors, you can copy and paste configuration details from a text file of the operating system chapters of this configuration guide. The format of this text file is similar to the printed version of the guide. Be sure to review the differences as explained at the beginning of the text file.

The `NetBackup_DeviceConfig_Guide.txt` file is installed with NetBackup server software in the following paths:

- `/usr/opensv/volmgr` (UNIX)
- `install_path\Veritas\Volmgr` (Windows)

The Hardware Compatibility List contains information about supported devices.

See [“About the NetBackup compatibility lists”](#) on page 10.

General device configuration sequence

Use the following general sequence when you configure devices:

- Physically connect the storage devices to the media server. Perform any hardware configuration steps that the device vendor or the operating system vendor specifies.
- Create any required system device files for the drives and robotic control. Device files are created automatically on Windows and on some UNIX platforms. Explicit configuration of device files is required on some UNIX servers to make full use of NetBackup features.

For SCSI controlled libraries, NetBackup issues SCSI commands to the robotic devices. SCSI commands allow NetBackup to discover and configure devices automatically. You may have to configure the server operating system to allow device discovery.

- Add the storage devices to NetBackup and configure them.
For instructions, see the [NetBackup Administrator's Guide, Volume I](#) or the **NetBackup Administration Console** help.
You can configure devices in NetBackup from the master server or the media server to which the devices are attached (the device host). For more information, see "To administer devices on other servers" in the [NetBackup Administrator's Guide, Volume I](#) or the **NetBackup Administration Console** help.

Configuration cautions

Observe the following cautions:

- In multiple-initiator (multiple host bus adapter) environments, NetBackup uses SCSI reservation to avoid tape drive usage conflicts and possible data loss problems. SCSI reservation operates at the SCSI target level; the hardware that bridges Fibre Channel to SCSI must work correctly.
By default, NetBackup uses SPC-2 SCSI reserve and release. Alternatively, you can use SCSI persistent reserve or disable SCSI reservation entirely.

For information about the NetBackup use of SCSI reservation, see the following:

- "Enable SCSI reserve" in the [NetBackup Administrator's Guide, Volume I](#).
- "How NetBackup reserves drives" in the [NetBackup Administrator's Guide, Volume II](#).
- Veritas does not recommend or support the use of single-ended to differential SCSI converters on NetBackup controlled devices. You may encounter problems if you use these converters.

About the NetBackup compatibility lists

Veritas provides compatibility lists for the operating systems, peripherals, and software with which NetBackup works.

See the NetBackup compatibility lists at the following webpage:

<http://www.netbackup.com/compatibility>

Operating systems

- [Chapter 2. AIX](#)
- [Chapter 3. HP-UX](#)
- [Chapter 4. Linux](#)
- [Chapter 5. Solaris](#)
- [Chapter 6. Windows](#)

AIX

This chapter includes the following topics:

- [Before you begin configuring NetBackup on AIX](#)
- [RS/6000 AIX adapter number conventions](#)
- [About AIX persistent naming support](#)
- [About configuring robotic control device files in AIX](#)
- [About device files for SAN Clients on AIX](#)
- [About configuring tape drive device files in AIX](#)
- [About Sony AIT drives on AIX](#)
- [AIX command summary](#)

Before you begin configuring NetBackup on AIX

Observe the following points when you configure the operating system:

- Verify that NetBackup supports your server platform and devices. Download the NetBackup hardware and operating system compatibility lists.
<http://www.netbackup.com/compatibility>
- Install and configure the IBM AIX Enhanced Device Driver (Atape device driver).
- Attach all peripherals and reboot the system before you configure the devices in NetBackup. When the computer is rebooted, AIX creates the device files for the attached peripherals.
- For many configuration steps, you can use the `smit` System Management Interface Tool. For more information, see the `smitt(1)` man page.

- To verify that the devices are configured correctly, use `smit` and `/usr/sbin/lssdev` command.
For the tape drives that you want to share among NetBackup hosts, ensure that the operating system detects the devices on the SAN before you configure the NetBackup Shared Storage Option.
- To obtain error and debug information about devices and robotic software daemons, the `syslogd` daemon must be active. See the AIX `syslogd(1)` man page for more information.

After you configure the hardware, add the robots and the drives to NetBackup.

RS/6000 AIX adapter number conventions

The location code for an adapter consists of two pairs of digits with the format AA-BB, as follows:

- AA identifies the location code of the drawer that contains the adapter card, as follows
 - If AA is 00, the adapter card is located in the CPU drawer or system unit, depending on the type of system.
 - If AA is not 00, the card is located in an I/O expansion drawer, as follows:
 - The first digit identifies the I/O bus; 0 corresponds to the standard I/O bus and 1 corresponds to the optional I/O bus.
 - The second digit identifies the slot on the I/O bus.
- BB identifies the I/O bus and the slot that contains the card, as follows:
 - The first digit of BB identifies the I/O bus that contains the adapter card, as follows:
 - If the card is in the CPU drawer or system unit, 0 represents the standard I/O bus and 1 represents the optional I/O bus. If the card is in an I/O expansion drawer, this digit is 0.
 - The second digit identifies the slot number on the I/O bus (or slot number in the I/O expansion drawer) that contains the card.

The following are examples of adapter numbers:

- 00-00 identifies the Standard I/O Planar.
- 00-05 indicates an adapter card that is in slot 5 of the standard I/O board. The board is in either the CPU drawer or the system unit, depending on the type of system.

- 00-12 indicates an adapter card that is in slot 2 of the optional I/O bus in the CPU drawer.
- 18-05 indicates an adapter card that is located in slot 5 of an I/O expansion drawer. The drawer is connected to the asynchronous expansion adapter that is located in slot 8 of the optional I/O bus in the CPU drawer.

About AIX persistent naming support

NetBackup requires that you enable persistent naming support for the AIX device files. Doing so ensures that the device targets and LUNs do not change after a system restart.

To enable persistent naming support, use the AIX SMIT utility or the `chdev` command to change the logical names of the devices. Change the logical names after the initial device configuration in AIX. For more information, see the IBM documentation.

About configuring robotic control device files in AIX

For IBM robotic libraries, Veritas supports the IBM AIX Enhanced Device Driver (Atape device driver) for use with NetBackup. NetBackup discovers the device files when you configure devices.

For information about the driver and how to configure device files, see the IBM documentation.

For robotic libraries other than IBM, Veritas recommends that you use an operating system other than AIX as the robotic control host.

About device files for SAN Clients on AIX

NetBackup SAN clients use tape drivers and SCSI pass-through methods for Fibre Transport traffic to NetBackup FT media servers. An AIX SAN Client that uses the standard tape driver can discover Fibre Transport targets on the FT media servers. The media server FT devices appear as `ARCHIVE Python` tape devices during SCSI inquiry from the SAN client. However, they are not tape devices and do not appear as tape devices in NetBackup device discovery.

During system startup, the AIX `cfgmgr` command configures all the devices that are necessary to use the system. If a NetBackup SAN Client cannot discover the FT devices, you can configure the device files on the client manually. Use the same procedure that you use for tape devices.

See [“Creating AIX no rewind device files for tape drives”](#) on page 17.

About configuring tape drive device files in AIX

The following topics provide information about configuring tape drive device files on AIX systems.

Table 2-1 AIX tape drive device file topics

Subject	Topic
About choosing a tape driver	See “About choosing a tape driver on AIX” on page 15.
About non-QIC tape drives	See “About non-QIC tape drives on AIX” on page 16.
About extended-file marks for drives	See “About extended-file marks for drives” on page 16.
About fast-tape positioning (locate-block) on AIX	See “About fast-tape positioning (locate-block) on AIX” on page 17.
Creating no rewind device files for tape drives	See “About no rewind device files on AIX” on page 17. See “Creating AIX no rewind device files for tape drives” on page 17. See “Example of creating a no rewind device file” on page 19.
About multiple tape densities	See “Using multiple tape densities on AIX” on page 21.
About SPC-2 SCSI reserve on AIX	See “About SPC-2 SCSI reserve on AIX” on page 21. See “Disabling SPC-2 SCSI reserve in AIX” on page 22.

About choosing a tape driver on AIX

If you use IBM tape drives, Veritas recommends that you install the IBM AIX Atape driver. For information about the driver, see the IBM documentation.

If you use other tape drives, Veritas recommends that you use the IBM AIX ost (other SCSI tape) driver. For information about the driver, see the IBM documentation.

For information about the drivers and how to configure device files, see the IBM documentation.

About non-QIC tape drives on AIX

Variable length block and fixed length block refer to how the operating system reads from and writes to a tape. Variable-mode devices allow more flexibility to read previously written tapes. Many tape devices can be accessed in either mode. NetBackup assumes variable length for non-quarter inch cartridge (QIC) drives.

For more information, see the `chdev(1)` and `smit(1)` man pages and the system management guide. The `smit` application is the most convenient way to change from fixed to variable-length-block devices manually.

Warning: For NetBackup, you must configure non-QIC tape drives as variable-length-block devices. Otherwise NetBackup can write data but may not be able to read it correctly. During a read, you may see a `not in tar format error`.

When you add a non-QIC tape drive to NetBackup, NetBackup issues the `chdev` command to configure the drive as a variable length block device. For reference, the following is the command that NetBackup uses to configure a drive for variable mode:

```
/usr/sbin/chdev -l Dev -a block_size=0
```

Dev represents the logical identifier for the drive (for example: `rmt0` or `rmt1`).

Therefore, you do not have to configure the drive manually for variable mode.

About extended-file marks for drives

If a tape drive supports extended file marks, you must configure it to use them (for example, 8-mm drives). Otherwise, NetBackup may not be able to use those drives.

For information, see the AIX `chdev(1)` and `smit(1)` man pages.

When you add a tape drive to NetBackup, NetBackup issues the `chdev` command to configure the drive to use extended file marks. For reference, the following is the command that NetBackup uses:

```
/usr/sbin/chdev -l Dev -a extfm=yes
```

Replace *Dev* with the logical identifier for the drive (such as `rmt0` or `rmt1`).

Therefore, you do not have to configure the drive manually for extended file marks.

About fast-tape positioning (locate-block) on AIX

Applies to AIT, DLT, Exabyte, and half-inch cartridge tape drives.

To position a tape to a specific block, NetBackup supports the SCSI `locate-block` command.

NetBackup uses the `locate-block` command by default.

Veritas recommends that you do not disable `locate-block` positioning. If you need to disable it, execute the following command:

```
touch /usr/openv/volmgr/database/NO_LOCATEBLOCK
```

With `locate-block` positioning disabled, NetBackup uses the forward-space-file/record method.

About no rewind device files on AIX

By default, NetBackup uses no rewind device files. These SCSI device files are in the `/dev` directory and have the following format:

```
/dev/rmtID.1
```

ID is the logical identifier assigned to the device by the system. The `.1` extension specifies the no rewind, no retention on open device file.

Normally, AIX creates tape drive device files automatically at boot time. Alternatively, you can run the AIX `cfgmgr` command, which should create the device files. If they do not exist, you must create them for the tape drives.

See [“Creating AIX no rewind device files for tape drives”](#) on page 17.

Creating AIX no rewind device files for tape drives

NetBackup uses no rewind device files for tape drives and for NetBackup SAN Clients. During system startup, the AIX `cfgmgr` command configures all the devices that are necessary to use the system. If necessary, you can use the following procedure to check for and create a no rewind device file.

To check for and create a no rewind device file

- 1 Display the I/O controllers in the system by using the following command:

```
/usr/sbin/lsdev -C | grep I/O
```

The following sample output shows that SCSI controller 1 (00-01) has been assigned the logical identifier `scsi0`.

```
scsi0 Available 00-01 SCSI I/O Controller
```

- 2 Display the SCSI and Fibre Channel devices in the system by using the following command. For SCSI devices, use `scsi` for the *type*; for Fibre Channel Protocol devices, use `fcp` for the *type*.

```
/usr/sbin/lsdev -C -s type
```

The following example shows two disk drives and a tape drive:

```
hdisk0 Available 00-01-00-0,0 400 MB SCSI Disk Drive
hdisk1 Available 00-01-00-1,0 400 MB SCSI Disk Drive
rmt0 Available 00-01-00-3,0 Other SCSI Tape Drive
```

If the device files for the tape drives exist, they appear in the output as `rmt0`, `rmt1`, and so on. The previous example output shows `rmt0`.

- 3 If a device file does not exist for the wanted tape drive, create it by using the following command:

```
/usr/sbin/mkdev -c tape -s scsi -t ost -p controller -w id,lun
```

The following are the arguments for the command:

- *controller* is the logical identifier of the drive's SCSI adapter, such as `scsi0`, `fscsi0`, or `vscsi1`.
- *scsi_id* is the SCSI ID of the drive connection.
- *lun* is the logical unit number of the drive connection.

For example, the following command creates a device file for a non-IBM 8-mm drive connected to controller `scsi0` at SCSI address 5,0:

```
mkdev -c tape -s scsi -t ost -p scsi0 -w 5,0
```

- 4 To verify, display the SCSI device files by using the `lsdev` command, as follows:

```
/usr/sbin/lsdev -C -s scsi
hdisk0 Available 00-01-00-0,0 400 MB SCSI Disk Drive
hdisk1 Available 00-01-00-1,0 400 MB SCSI Disk Drive
rmt0 Available 00-01-00-3,0 Other SCSI Tape Drive
rmt1 Available 00-01-00-5,0 Other SCSI Tape Drive
```

The output shows that the `rmt1` device file was created.

- 5 If the device files do not exist on an FCP controller, use the following command to create them:

```
/usr/sbin/cfgmgr -l device
```

`device` is the controller number from step 1.

- 6 Ensure that the device is configured for variable-mode and extended file marks. Use the `chdev` command, as follows (`dev` is the logical identifier for the drive (for example, `rmt1`)).

```
/usr/sbin/chdev -l dev -a block_size=0
/usr/sbin/chdev -l dev -a extfm=yes
```

- 7 To configure the drive manually in NetBackup, enter the following device file pathname:

```
/dev/rmt1.1
```

Example of creating a no rewind device file

This topic provides an example of how to create a no rewind device file on AIX for NetBackup. Assume the device files for the wanted SCSI 8-mm tape drive (controller 1, SCSI ID 5) do not exist.

To create a no rewind device file for SCSI ID 5

- 1 Determine the logical identifier for the SCSI controller as follows:

```
/usr/sbin/lsdev -C -c adapter | grep SCSI
```

The following output shows that scsi0 is the logical name for SCSI controller 1:

```
scsi0 Available 00-01 SCSI I/O Controller
```

- 2 Determine if the device files exist for any device at SCSI ID 5.

```
/usr/sbin/lsdev -C -s scsi
```

The following output shows that some device files exist for tape and disk. However, a device files does not exist for the 8-mm tape drive at controller 1 (scsi0) and SCSI ID 5 (5,0):

```
hdisk0 Available 00-01-00-0,0 400 MB SCSI Disk Drive
hdisk1 Available 00-01-00-1,0 400 MB SCSI Disk Drive
rmt0 Available 00-01-00-3,0 Other SCSI Tape Drive
```

- 3 Create the device file by using the following command:

```
mkdev -c tape -t ost -s scsi -p scsi0 -w 5,0
```

- 4 Display the device files by issuing the following command:

```
/usr/sbin/lsdev -C -s scsi
hdisk0 Available 00-01-00-0,0 400 MB SCSI Disk Drive
hdisk1 Available 00-01-00-1,0 400 MB SCSI Disk Drive
rmt0 Available 00-01-00-3,0 Other SCSI Tape Drive
rmt1 Available 00-01-00-5,0 Other SCSI Tape Drive
```

- 5 To ensure that the tape device is configured for variable-mode and extended file marks, use the following commands:

```
chdev -l rmt1 -a block_size=0
chdev -l rmt1 -a extfm=yes
```

- 6 To configure the drive manually in NetBackup, enter the following device file pathname:

```
/dev/rmt1.1
```

Using multiple tape densities on AIX

After you create the device files for your tape drives, you can configure densities on the drives that support multiple densities. Exabyte 8500C is an example of a tape drive for which you can use different densities.

AIX supports two configuration settings for density, although not all tape drives support multiple densities. The default density for both density setting 1 and density setting 2 is 0, which means maximum density.

The following procedure is an example of using the `chdev` command to modify the density settings. Alternatively, you can use the System Manager Interface Tool (SMIT).

To modify the density settings

- ◆ The following commands modify both of the tape drive device files:

```
chdev -l tapedev -a density_set_1=density
```

```
chdev -l tapedev -a density_set_2=density
```

The following describe the arguments to the command options:

- *tapedev* is the logical identifier for the drive, such as `rmt0` or `rmt1`.
- *density* is a decimal number from 0 to 255 that represents the wanted density. Zero (0) selects the default density for the tape drive, which is usually the drive's high density setting. Permitted values and their meanings vary with different types of tape drives.

To use density setting 1, use the following no rewind on close device file when you configure the device in NetBackup:

```
/dev/rmt*.1
```

To use density setting 2, use the following no rewind on close device file when you configure the device in NetBackup:

```
/dev/rmt*.5
```

About SPC-2 SCSI reserve on AIX

By default, NetBackup uses SPC-2 SCSI reserve and release for tape drive reservations in shared drive environments. The NetBackup Shared Storage Option provides shared drive functionality in NetBackup.

Alternatively, you can use SCSI persistent reserve for shared tape drive reservations in NetBackup, as follows:

- For the tape drives that support SPC-3 Compatible Reservation Handling (CRH), you can use SCSI persistent reserve by enabling it in NetBackup. No special configuration in AIX is required.
- For the tape drives that do not support CRH, you must disable SPC-2 SCSI reserve in AIX for those drives. After you disable SPC-2 SCSI reserve, you can use persistent reserve by enabling it in NetBackup. If the drive does not support CRH and you do not disable SPC-2 SCSI reserve, access attempts to the drive fail.

See “[Disabling SPC-2 SCSI reserve in AIX](#)” on page 22.

Warning: If the tape driver does not let you disable SPC-2 SCSI reserve, do not use SCSI persistent reserve with the drives that do not support CRH. The sctape driver is an example of a tape driver that lets you disable SPC-2 SCSI reserve.

For more information about NetBackup and SCSI reservations, see the following:

- The description of the **Enable SCSI Reserve Media** host property in the *NetBackup Administrator's Guide, Volume I*.
<http://www.veritas.com/docs/DOC5332>
- The "How NetBackup reserves drives" topic in the *NetBackup Administrator's Guide, Volume II*.
<http://www.veritas.com/docs/DOC5332>

Disabling SPC-2 SCSI reserve in AIX

To disable SPC-2 SCSI reserve, use the AIX `chdev` command to change the reserve attribute for the device file for the tape drive.

For information about the `chdev` command, see the AIX `chdev` man page.

To disable SPC-2 SCSI reserve in AIX

- ◆ Invoked the following command:

```
chdev -l name -a res_support=no
```

Replace *name* with the name of the device file, such as `rmt0`.

About Sony AIT drives on AIX

Sony S-AIT drives have DIP switches (SWA and SWB) located on the bottom of the drive.

For the NetBackup Shared Storage Option, you must set the SWA-1 (**No Reply Unit Attention After Process Login**) switch correctly. Its factory setting depends on the serial number of the drive.

If the drive has an early serial number, switch SWA-1 may be OFF. If so, change the switch to ON.

Drives with newer serial numbers have SWA-1 set to ON as the default.

These newer serial numbers are as follows:

- SDZ-130 :01442007 and later
- SDZ-130/L :01200696 and later

Also, for the drives that have a date of May 17, 2004 and later, the DIP switch is set to ON.

The following table shows the dip switch settings for the newer serial number drives.

Table 2-2 DIP switch positions for AIT drives

Switch	Setting (1 = ON and 0 = OFF)
SWA-1	1
SWA-2	0
SWA-3	0
SWA-4	0
SWA-5	0
SWA-6	0
SWA-7	1
SWA-8	0

AIX command summary

The following is a summary of commands that may be useful when you configure devices. For examples of their usage, see the procedures in this chapter.

- `/usr/sbin/lsdev -C | grep I/O`
Displays the adapters that are physically available on your server.
- `/usr/sbin/lsdev -C -s filetype`
Displays the device files that have been created, where *filetype* defines the type of file displayed: `scsi` displays SCSI files and `fc` displays Fibre Channel files.

- `mkdev -c tape -s scsi -t ost -p controller -w id,lun`
Creates the device files for tapes.
controller is the logical identifier of the drive SCSI adapter (such as `scsi0` or `scsi1`). *id* is the SCSI ID of the robotic connection. *lun* is the logical unit number of the robotic connection.
- `/usr/sbin/chdev -l dev -a block_size=0`
Configures the drive with logical identifier specified by *dev* (for example: `rmt0`) to variable mode.
- `/usr/sbin/chdev -l dev -a extfm=yes`
Configures the drive with logical identifier specified by *dev* (for example: `rmt0`) for extended file marks.
- `/etc/lstat -l dev -E -H`
Displays the device information, where *dev* is the name of the device (for example, `rmt1`).
- `/usr/sbin/cfgmgr -l device`
Creates the device files on a Fibre Channel Protocol controller, where *device* is the controller number (for example, `fscsi0`).
- `/usr/bin/odmget -q "name=rmtX" CuAt`
Displays the device attributes for the device (*rmtX*). This command can be used to determine SCSI target and LUN pairs when you configure Fibre Channel devices.
Where *rmtX* is the name of the tape device (for example: `rmt0` or `rmt1`).

HP-UX

This chapter includes the following topics:

- [Before you begin configuring NetBackup on HP-UX](#)
- [About robotic control on HP-UX](#)
- [About HP-UX device addressing schemes](#)
- [HP-UX tape drive device file requirements for NetBackup](#)
- [About device drivers and files for HP-UX persistent DSFs](#)
- [About configuring persistent DSFs](#)
- [About HP-UX legacy device drivers and files](#)
- [Creating device files for SAN Clients on HP-UX](#)
- [About configuring legacy device files](#)
- [About SPC-2 SCSI reserve on HP-UX](#)
- [Disabling SPC-2 SCSI reserve in HP-UX](#)
- [About disabling the HP-UX EMS Tape Device Monitor for a SAN](#)
- [HP-UX command summary](#)

Before you begin configuring NetBackup on HP-UX

Observe the following points when you configure the operating system:

- Verify that NetBackup supports your server platform and devices. Download the NetBackup hardware and operating system compatibility lists.

<http://www.netbackup.com/compatibility>

- For SCSI controlled libraries, NetBackup issues SCSI commands to the robotic devices. For NetBackup to function correctly, the properly named device files must exist.
- To verify that the devices are configured correctly, use the HP-UX `sam` utility and the `ioscan -f` command.
 For the tape drives that you want to share among NetBackup hosts, ensure that the operating system detects the devices on the SAN before you configure the NetBackup Shared Storage Option.
- Some HP SCSI adapters do not support SCSI pass through; therefore, devices on such adapters are not discovered automatically.

After you configure the hardware, add the robots and the drives to NetBackup.

About robotic control on HP-UX

Several options exist for robotic control, as follows:

- SCSI, Serial-attached SCSI (SAS), or Fibre Channel Protocol control.
 SCSI control includes Fibre Channel Protocol (FCP), which is SCSI over Fibre Channel. A robotic device in a library moves the media between storage slots and the drives in the library.
 See [“About HP-UX device addressing schemes”](#) on page 26.
- API control over a LAN.
 See the "ADIC Automated Media Library (AML)" topic.
 See the "IBM Automated Tape Library (ATL)" topic.
 See the "Oracle Sun StorageTek ACSLS robots" topic.
 If you use API control, you still must configure tape drive device file access in HP-UX.

About HP-UX device addressing schemes

NetBackup supports the following two HP-UX device addressing schemes for mass storage devices:

- Agile addressing, which was introduced in HP-UX 11i v3. Agile addressing uses persistent device special files (DSFs) for devices.
 NetBackup device discovery finds persistent DSFs only. Therefore, Veritas recommends that you use persistent DSFs.
 See [“About device drivers and files for HP-UX persistent DSFs”](#) on page 28.
 See [“About configuring persistent DSFs”](#) on page 30.

- The legacy naming model.
 Legacy device files are supported in HP-UX 11i v3 and earlier.
 If you use the legacy model, you must configure devices manually in NetBackup.
 You cannot use NetBackup device discovery and automatic configuration.
 See [“About HP-UX legacy device drivers and files”](#) on page 32.
 See [“About configuring legacy device files”](#) on page 35.

When HP-UX 11i v3 is installed, both legacy and persistent DSFs are created on the system. Both types of DSFs can coexist and may be used simultaneously to access mass storage devices.

NetBackup requires specific device file capabilities for tape drives.

See [“HP-UX tape drive device file requirements for NetBackup”](#) on page 27.

You may be required to perform other HP-UX configuration.

See [“About SPC-2 SCSI reserve on HP-UX”](#) on page 47.

See [“Creating device files for SAN Clients on HP-UX”](#) on page 35.

See [“About disabling the HP-UX EMS Tape Device Monitor for a SAN”](#) on page 47.

See [“Disabling SPC-2 SCSI reserve in HP-UX”](#) on page 47.

HP-UX tape drive device file requirements for NetBackup

The following table describes the requirements for tape drive device files.

Table 3-1 Tape drive device file requirements

Requirement	Description
Berkeley-style close	<p>NetBackup requires Berkeley-style close for tape drive device files. The letter b in the file name indicates Berkeley-style close device files.</p> <p>In Berkeley-style close, the tape position remains unchanged by a device close operation. (Conversely, in AT&T-style close, the drive advances the tape to immediately after the next end-of-file (EOF) marker.) To establish the correct position for the next tape operation, applications must assume the tape's position after a close. NetBackup assumes Berkeley-style close on HP-UX systems.</p>

Table 3-1 Tape drive device file requirements (*continued*)

Requirement	Description
Fast-tape positioning (locate-block)	<p>HP-UX supports locate-block for most drive types for Fast Wide GSC SCSI adapters. For a list of drive types that NetBackup supports, see the Veritas Hardware Compatibility List:</p> <p>http://www.netbackup.com/compatibility</p> <p>NetBackup uses the locate-block command by default if a pass-through path is configured.</p> <p>To disable locate-block positioning, execute the following command:</p> <pre>touch /usr/opensv/volmgr/database/NO_LOCATEBLOCK</pre> <p>When locate-block positioning is disabled, NetBackup uses the forward-space-file/record method.</p>
No rewind on close	<p>NetBackup requires no rewind on close for tape devices. The letter <i>n</i> in the file name indicates no rewind device files.</p>

About device drivers and files for HP-UX persistent DSFs

NetBackup supports persistent device special files (DSFs) for robotic tape libraries and tape drives. Persistent DSFs are a component of a new storage stack addressing model HP calls agile addressing. The device World Wide Identifier (WWID) identifies a device; a device path does not identify a device. The device file name of the logical unit is independent of the paths that lead to the LUN.

When HP-UX 11i v3 is installed, both persistent DSFs and legacy device files are created on the system.

You can use persistent DSFs on some HP-UX servers and legacy device files on others. However, if you use legacy device files, you must configure devices manually in NetBackup.

Note: NetBackup device discovery finds persistent DSFs only. Therefore, Veritas recommends that you use persistent DSFs.

See [“About legacy pass-through paths for tape drives”](#) on page 34.

See [“About legacy tape drive device files”](#) on page 33.

About device drivers for persistent DSFs

The following are the device drivers required to use persistent DSFs:

- The `eschgr` driver for robotic control.
- The `estape` driver for tape drives.
- For IBM tape drives, the `atdd` driver. NetBackup requires a minimum `atdd` driver level. See the Hardware Compatibility List for IBM `atdd` driver levels that are supported and `atdd` configuration information.

<http://www.veritas.com/docs/TECH76495>

Also see the Operating System Compatibility List for the minimum OS patch level that is required to run on HP-UX:

<http://www.veritas.com/docs/TECH76648>

About persistent DSFs for robotic control

The following is the persistent DSF name format for robotic control:

```
/dev/rchgr/autoch#
```

The # represents the instance number. For example, if HP-UX discovers two robotic devices (auto-changers) and assigns them instance numbers 0 and 1 respectively, HP-UX automatically creates the following device files:

```
/dev/rchgr/autoch0
```

```
/dev/rchgr/autoch1
```

About persistent DSFs for tape drive access

The following is the persistent DSF name format for tape drive read and write access:

```
/dev/rtape/tape#_BESTnb
```

The following describe the DSF name format:

- The # represents the instance number.
- BEST indicates the highest density.
- n indicates no rewind on close.
- b indicates Berkeley-style close.

About persistent DSF pass-through paths

Although NetBackup requires the `/dev/rtape` DSFs to configure tape drives, NetBackup uses pass-through device files for drive access.

NetBackup creates pass-through paths for all valid `/dev/rtape` paths. NetBackup creates the paths during device discovery or when you run the `/usr/opensv/volmgr/bin/scan` command. The following is the file name format:

```
/dev/pt/pt_tape#
```

The # represents the instance number that matches the number in the `/dev/rtape/tape#_BESTnb` device file or that is retrieved from the `ioscan` output.

Although NetBackup uses the pass-through device files during tape drive operations, you specify the `/dev/rtape` device files if you configure the drives manually in NetBackup. NetBackup then uses the appropriate pass-through device files.

See [“Creating persistent DSF pass-through paths”](#) on page 32.

About configuring persistent DSFs

Configure persistent DSFs for the following:

- Robotic control.
 - See [“Creating HP-UX persistent DSFs”](#) on page 30.
 - See [“Upgrading NetBackup to use HP-UX persistent DSFs”](#) on page 31.
- Tape drive read and write accesss.
 - See [“Creating HP-UX persistent DSFs”](#) on page 30.
 - See [“Creating persistent DSF pass-through paths”](#) on page 32.

Creating HP-UX persistent DSFs

By default, new installations of HP-UX 11i v3 and later create both persistent DSFs and legacy device files for the `eschgr` and `estape` drivers. However, you can reinstall or create the persistent DSFs.

First connect the devices to the system. Then, create persistent DSFs on every server on which you want to use the new persistent DSFs.

To create the persistent DSFs automatically

- ◆ Depending on the driver, enter the following command as root:
 - For the `eschgr` autochanger driver, enter `insf -d eschgr`.
 - For the `estape` tape driver, enter `insf -d estape`.
 - For the IBM `atdd` tape driver, enter `insf -d atdd`
- To refresh the device path for all devices that use a driver, adding the `-e` option to the command line.
- For more information about using the HP-UX `insf` command, see the man page.

Upgrading NetBackup to use HP-UX persistent DSFs

Use the following procedure to configure an existing NetBackup environment to use persistent DSFs after you upgrade a media server to HP-UX 11i v3.

The following procedure also removes the legacy paths from the NetBackup device configuration. To save the legacy paths, you can use the NetBackup `tpconfig` utility to disable the old paths but retain them in the device configuration.

If you run the NetBackup **Device Configuration Wizard** before you perform this change, NetBackup adds the new DSF paths to the device configuration. However, the legacy paths remain configured as alternate paths.

To change a NetBackup media server to use the new DSFs

- 1 Add the following entry to the `/usr/openv/volmgr/vm.conf` file (the syntax must be as specified in all capital letters):

```
AUTO_PATH_CORRECTION=YES
```

The `AUTO_PATH_CORRECTION` entry directs NetBackup to scan for device paths when the `ltid` device daemon starts.

- 2 After adding the entry to the `vm.conf` file and with no current jobs running on the media server, run the following command:

```
/usr/openv/volmgr/bin/stopltd
```

- 3 Wait a few minutes for the service to stop, then restart `ltid` by running the following command:

```
/usr/opensv/volmgr/bin/ltid
```

Upon start-up, `ltid` scans for device paths, adds the new DSFs, and then purges the legacy DSFs from your NetBackup configuration for the media server.

After `ltid` starts, only the new persistent DSF paths should be configured in NetBackup.

- 4 After the services are started and device paths updated, you can (but do not have to) remove the `AUTO_PATH_CORRECTION=YES` entry from the `vm.conf` file.

Creating persistent DSF pass-through paths

NetBackup creates pass-through paths for all valid `/dev/rtape` paths. NetBackup creates the paths during device discovery or when you run the

```
/usr/opensv/volmgr/bin/scan
```

 command.

However, you can create them manually. The following example shows how to create a pass-through device file for a persistent DSF.

To create a pass-through path for a tape drive

- ◆ Enter the following command (`#` is the instance number of the device from the `ioscan` output):

```
mksf -P -C tape -I #
```

For more information about using the HP-UX `mksf` command, see the man page.

About HP-UX legacy device drivers and files

Legacy device files are the old style device files that are not part of the HP-UX agile addressing scheme.

NetBackup does not discover legacy device files. If you use legacy device files, you must configure devices manually in NetBackup.

Note: NetBackup device discovery finds persistent DSFs only. Therefore, Veritas recommends that you use persistent DSFs.

See [“About device drivers and files for HP-UX persistent DSFs”](#) on page 28.

About HP-UX device drivers for legacy device files

The following are the drivers supported:

- The `sctl` driver for robotic control.
- The `stape` driver for tape drives.
- For IBM tape drives, the `atdd` driver. NetBackup requires a minimum `atdd` driver level. See the Hardware Compatibility List for IBM `atdd` driver levels that are supported and `atdd` configuration information.
<http://www.netbackup.com/compatibility>
 NetBackup also supports the use of the IBM `atdd` tape driver on HP-UX 11i v3. Also see the Operating System Compatibility List for the minimum OS patch level that is required to run on HP-UX:
<http://www.netbackup.com/compatibility>

About legacy robotic control device files

For SCSI robotic control, NetBackup can use the `/dev/sctl` device files. The device file names have the following format:

```
/dev/sctl/cCARDtTARGETlLUN c Major 0xIIITL00
```

Where:

- `CARD` is the card instance number of the adapter.
- `TARGET` is the SCSI ID of the robotic control.
- `LUN` is the SCSI logical unit number (LUN) of the robot.
- `Major` is the character major number (from the `lsdev` command).
- `II` are two hexadecimal digits that represent the card instance number.
- `T` is a hexadecimal digit that represents the SCSI ID of robotic control.
- `L` is a hexadecimal digit that represents the SCSI LUN of the robotic control.

A library may have more than one robotic device. Each robotic device requires a device file.

See “[Creating legacy SCSI and FCP robotic controls on HP-UX](#)” on page 36.

About legacy tape drive device files

NetBackup requires the `/dev/rmt` device files to configure tape drives.

The device file names have the following format:

```
/dev/rmt/c#t#d#BESTnb
```

The following describe the device file names:

- `c#` is the card instance number.
- `t#` is the SCSI ID.
- `d#` is the device LUN.
- `BEST` indicates the highest density format and data compression the device supports.
- `n` indicates no rewind on close.
- `b` indicates Berkeley-style close.

The following are examples of tape drive device files:

```
/dev/rmt/c7t0d0BESTnb  
/dev/rmt/c7t1d0BESTnb  
/dev/rmt/c7t4d0BESTnb  
/dev/rmt/c7t5d0BESTnb
```

See [“About creating legacy tape drive device files”](#) on page 43.

About legacy pass-through paths for tape drives

Although NetBackup requires the `/dev/rmt` device files to configure tape drives, NetBackup uses pass-through device files for drive access.

On media servers, NetBackup automatically creates pass-through device files if the appropriate `/dev/rmt` tape drive device files exist. NetBackup creates the pass-through device files in the `/dev/sct1` directory.

NetBackup does not modify or delete any existing pass-through paths.

NetBackup does not detect the type of adapter cards that are installed in the system. Therefore, NetBackup creates pass-through paths for tape drives connected to the adapter cards that do not support pass through. These pass-through paths do not cause problems.

Although NetBackup uses the pass-through device files during tape drive operations, you specify the `/dev/rmt` device files when you configure the drives in NetBackup. NetBackup then uses the appropriate pass-through device files.

Usually, you do not have to create pass-through paths for drives. However, instructions to do so are provided for reference.

NetBackup SAN clients require legacy pass-through device files.

See [“Creating device files for SAN Clients on HP-UX”](#) on page 35.

Note: Pass-through paths are not supported on HP-PB adapters such as HP28696A - Wide SCSI or HP 28655A - SE SCSI.

See [“Creating tape drive pass-through device files”](#) on page 43.

See [“About device drivers and files for HP-UX persistent DSFs”](#) on page 28.

Creating device files for SAN Clients on HP-UX

NetBackup SAN clients use tape drivers and SCSI pass-through methods for Fibre Transport traffic to NetBackup FT media servers. On HP-UX systems, NetBackup SAN clients require the `sctl` driver and pass-through tape drive device files.

The following table describes the tasks that create the device files. Before you create the device files, the NetBackup FT media server must be active and the SAN must be zoned correctly as described in the [NetBackup SAN Client and Fibre Transport Guide](#).

Table 3-2 SAN Client device file tasks

Step	Action	Description
Step 1	If the <code>sctl</code> driver is not the default pass-through driver on your system, install and configure the <code>sctl</code> driver.	See the HP-UX <code>scsi_ctl(7)</code> man page.
Step 2	Create the pass-through paths required.	See “About legacy pass-through paths for tape drives” on page 34. See “Creating tape drive pass-through device files” on page 43.

The media server FT devices appear as `ARCHIVE Python` tape devices during SCSI inquiry from the SAN client. However, they are not tape devices and do not appear as tape devices in NetBackup device discovery.

About configuring legacy device files

You can use legacy device files for the following:

- Robotic control using SCSI or Fibre Channel Protocol control.
 SCSI control includes Fibre Channel Protocol (FCP), which is SCSI over Fibre Channel. A robotic device in a library moves the media between storage slots and the drives in the library.
 See [“Creating legacy SCSI and FCP robotic controls on HP-UX”](#) on page 36.

- Tape drive read and write access.
 See [“About creating legacy tape drive device files”](#) on page 43.
 See [“Creating tape drive pass-through device files”](#) on page 43.
- SAN client pass-through paths for Fibre Transport traffic to NetBackup media servers.
 See [“Creating device files for SAN Clients on HP-UX”](#) on page 35.

Creating legacy SCSI and FCP robotic controls on HP-UX

You must create the robotic control device files for the `sctl` driver manually; they are not created automatically when the system boots.

Before you create the device files, you must do the following:

- Install and configure the `sctl` driver. For more information, see the HP-UX `scsi_ctl(7)` man page.
 The `sctl` driver may be the default pass-through driver on your system. If so, you do not have to configure the kernel to use the `sctl` pass-through driver.
- Install and configure the `schgr` device driver. For more information, see the HP-UX `autochanger(7)` man page.
- Attach the devices.

Examples of how to create the device files are available.

See [“Example of how to create a sctl device file for SCSI \(PA-RISC\)”](#) on page 37.

See [“Example of how to create a sctl device file for FCP \(PA-RISC\)”](#) on page 39.

See [“Example of how to create sctl device files for FCP \(Itanium\)”](#) on page 41.

To create `sctl` device files

- 1 Invoke the `ioscan -f` command to obtain SCSI bus and robotic control information.
- 2 Examine the output for the card instance number and the SCSI ID and LUN of the robotic device, as follows:
 - The instance number of the card is in the I column of the output.
 - The `H/W Path` column of the changer output (`schgr`) includes the SCSI ID and LUN. Use the card's `H/W Path` value to filter the changer's `H/W Path` entry; the SCSI ID and the LUN remain.

- 3 Determine the character major number of the `sctl` driver by using the following command:

```
lsdev -d sctl
```

Examine the output for an entry that shows `sctl` in the Driver column.

- 4 Use the following commands to create the device file for the SCSI robotic control:

```
mkdir /dev/sctl  
cd /dev/sctl  
/usr/sbin/mknod cCARDtTARGETlLUN c Major 0xIITL00
```

Where:

- *CARD* is the card instance number of the adapter.
- *TARGET* is the SCSI ID of the robotic control.
- *LUN* is the SCSI logical unit number (LUN) of the robot.
- *Major* is the character major number (from the `lsdev` command).
- *II* are two hexadecimal digits that represent the card instance number.
- *T* is a hexadecimal digit that represents the SCSI ID of robotic control.
- *L* is a hexadecimal digit that represents the SCSI LUN of the robotic control.

Example of how to create a `sctl` device file for SCSI (PA-RISC)

In this example, the following robots exist:

- An ADIC Scalar 100 library is on a SCSI bus with an instance number of 7, SCSI ID 2, and LUN 0.
- The robotic control for an IBM ULT3583-TL library is on the same SCSI bus at SCSI ID 3 and LUN 0.

To create SCSI robotic device files for HP-UX PA-RISC

1 Invoke the `ioscan -f` command, as follows:

```
ioscan -f
Class      I  H/W Path      Driver  S/W State H/W Type  Description
=====
ext_bus    7  0/7/0/1       c720    CLAIMED  INTERFACE SCSI C896 Fast Wide LVD
target    10 0/7/0/1.0      tgt     CLAIMED  DEVICE
tape      65 0/7/0/1.0.0    stape   CLAIMED  DEVICE    QUANTUM SuperDLT1
target    11 0/7/0/1.1      tgt     CLAIMED  DEVICE
tape      66 0/7/0/1.1.0    stape   CLAIMED  DEVICE    QUANTUM SuperDLT1
target    12 0/7/0/1.2      tgt     CLAIMED  DEVICE
autoch    14 0/7/0/1.2.0    schgr   CLAIMED  DEVICE    ADIC Scalar 100
target    13 0/7/0/1.3      tgt     CLAIMED  DEVICE
autoch    19 0/7/0/1.3.0    schgr   CLAIMED  DEVICE    IBM ULT3583-TL
target    14 0/7/0/1.4      tgt     CLAIMED  DEVICE
tape      21 0/7/0/1.4.0    atdd    CLAIMED  DEVICE    IBM ULT3580-TD1
target    15 0/7/0/1.5      tgt     CLAIMED  DEVICE
tape      19 0/7/0/1.5.0    atdd    CLAIMED  DEVICE    IBM ULT3580-TD1
```

2 Examine the output for the card instance number and the SCSI ID and LUN of the robotic device, as follows:

The card H/W Path is 0/7/0/1; the card instance number (I column) is 7. Apply the H/W Path value as a mask. The ADIC robotic device (`schgr`) is at SCSI ID 2, LUN 0 on this bus. The IBM robotic device (`schgr`) is at SCSI ID 3, LUN 0 on this bus.

- 3** Determine the character major number of the `sctl` driver by using the following command:

```
lsdev -d sctl
Character      Block      Driver      Class
  203          -1         sctl        ctl
```

The output from this command shows that the character major number for the `sctl` driver is 203.

- 4** The commands to create the device files follow. For the ADIC robot, the card instance number is 7, the target is 2, and the LUN is 0. For the IBM robot, the card instance number is 7, the SCSI ID is 3, and the LUN is 0.

```
cd /dev/sctl
/usr/sbin/mknod c7t210 c 203 0x072000
/usr/sbin/mknod c7t310 c 203 0x073000
```

If you add the robots to NetBackup manually, you specify the following for ADIC robotic control and IBM robotic control respectively:

```
/dev/sctl/c7t210
/dev/sctl/c7t310
```

Example of how to create a sctl device file for FCP (PA-RISC)

The following example shows how create a `sctl` device file for an HP VLS9000 robot. NetBackup uses the device file for robotic control.

To create an FCP robotic device file for HP-UX PA-RISC

- 1** Invoke the `ioscan -f` command. The following output example is edited for readability:

```
ioscan -f
Class  I  H/W Path                Driver  S/W State  H/W Type  Description
=====
fc      0  0/2/0/0                td      CLAIMED   INTERFACE HP Tachyon XL2 Fibre
                                Channel Mass Storage
                                Adapter
fcp     4  0/2/0/0.10            fcp     CLAIMED   INTERFACE FCP Domain
ext_bus 6  0/2/0/0.10.11.255.0    fcpdev  CLAIMED   INTERFACE FCP Device Interface
target  5  0/2/0/0.10.11.255.0.0  tgt     CLAIMED   DEVICE
autoch  2  0/2/0/0.10.11.255.0.0.0 schgr   CLAIMED   DEVICE      HP      VLS
tape    5  0/2/0/0.10.11.255.0.0.1 stape   CLAIMED   DEVICE      HP      Ultrium 4-SCSI
tape    6  0/2/0/0.10.11.255.0.0.2 stape   CLAIMED   DEVICE      HP      Ultrium 4-SCSI
tape    7  0/2/0/0.10.11.255.0.0.3 stape   CLAIMED   DEVICE      HP      Ultrium 4-SCSI
```

- 2** Examine the output for the card instance number and the SCSI ID and LUN of the robotic device. In this example, the interface card instance number (the `I` column) is 6. If you use the card's `H/W Path` value as a mask (`0/2/0/0.10.11.255.0`), you see the following:
 - An HP VLS9000 robot is at SCSI ID 0, LUN 0.
 - Three Ultrium 4-SCSI drives are at SCSI ID 0 and LUN 1, LUN 2, and LUN 3.

- 3** Determine the character major number of the `sctl` driver by using the `lsdev` command, as follows:

```
lsdev -d sctl
Character      Block      Driver      Class
  203          -1         sctl        ctl
```

The output from this command shows that the character major number for the `sctl` driver is 203.

- 4** The commands to create the device file for the HP VLS9000 robotic control are as follows. The card instance number is 6, the target is 0, and the LUN is 0.

```
cd /dev/sctl
/usr/sbin/mknod c6t010 c 203 0x060000
```

If you add the robot to NetBackup manually, specify the following pathname for robotic control:

```
/dev/sctl/c6t010
```

Example of how to create `sctl` device files for FCP (Itanium)

With Fibre Channel, the hardware paths are longer than with SCSI.

In this example, the following devices are attached to the host.

- An HP EML E-Series robot with four HP drives (two LTO2 and two LTO3 drives). A separate path exists for each drive pair. The robotic control is through card instance 12 (0/4/1/1.2.12.255.0).
- An HP VLS 6000 robot with six drives. The robot is partitioned into two virtual libraries, three Quantum SDLT320 drives in one library and three HP LTO3 drives in the other library. Separate robotic control exists for each library.

To create FCP robotic device files for HP-UX Itanium

- 1 Invoke the `ioscan -f` command. The following is a command output excerpt that shows the Fibre Channel devices on a host:

```

ext_bus  4  0/4/1/1.2.10.255.0      fcd_vbus CLAIMED INTERFACE FCP Device Interface
target   7  0/4/1/1.2.10.255.0.0      tgt       CLAIMED DEVICE
tape     18 0/4/1/1.2.10.255.0.0.0    stape    CLAIMED DEVICE      HP Ultrium 3-SCSI
tape     20 0/4/1/1.2.10.255.0.0.1    stape    CLAIMED DEVICE      HP Ultrium 3-SCSI
ext_bus  13  0/4/1/1.2.11.255.0      fcd_vbus CLAIMED INTERFACE FCP Device Interface
target   8  0/4/1/1.2.11.255.0.0      tgt       CLAIMED DEVICE
autoch   4  0/4/1/1.2.11.255.0.0.0    schgr    CLAIMED DEVICE      HP VLS
tape     22 0/4/1/1.2.11.255.0.0.1    stape    CLAIMED DEVICE      QUANTUM SDLT320
tape     23 0/4/1/1.2.11.255.0.0.2    stape    CLAIMED DEVICE      QUANTUM SDLT320
tape     24 0/4/1/1.2.11.255.0.0.3    stape    CLAIMED DEVICE      QUANTUM SDLT320
autoch   5  0/4/1/1.2.11.255.0.0.4    schgr    CLAIMED DEVICE      HP VLS
tape     25 0/4/1/1.2.11.255.0.0.5    stape    CLAIMED DEVICE      HP Ultrium 3-SCSI
tape     26 0/4/1/1.2.11.255.0.0.6    stape    CLAIMED DEVICE      HP Ultrium 3-SCSI
tape     27 0/4/1/1.2.11.255.0.0.7    stape    CLAIMED DEVICE      HP Ultrium 3-SCSI
ext_bus  12  0/4/1/1.2.12.255.0      fcd_vbus CLAIMED INTERFACE FCP Device Interface
target   6  0/4/1/1.2.12.255.0.0      tgt       CLAIMED DEVICE
autoch   1  0/4/1/1.2.12.255.0.0.0    schgr    CLAIMED DEVICE      HP EML E-Series
tape     19 0/4/1/1.2.12.255.0.0.1    stape    CLAIMED DEVICE      HP Ultrium 2-SCSI
tape     21 0/4/1/1.2.12.255.0.0.2    stape    CLAIMED DEVICE      HP Ultrium 2-SCSI
  
```

- 2 Examine the output for the card instance number and the SCSI ID and LUN of the robotic device.

In this example, the following devices are attached to this host:

- The robotic control for the HP EML E-Series robot is through card instance 12 (0/4/1/1.2.12.255.0). Two of the drives are accessed through the same path, and the other two are accessed through card instance 4 (0/4/1/1.2.10.255.0).
- The robotic controls for the HP VLS 6000 robot partitions are through card instance 13. Robotic control for one partition is at SCSI ID 0 and LUN 0. Robotic control for the other partition is at SCSI ID 0 and LUN 4.

- 3 Determine the character major number of the `sctl` driver by using the following command:

```
lsdev -d sctl
Character      Block      Driver      Class
   203         -1         sctl        ctl
```

The output from this command shows that the character major number for the `sctl` driver is 203.

- 4 The commands to create the devices file for the robotic controls are as follows:

```
cd /dev/sctl
/usr/sbin/mknod c12t010 c 203 0x0c0000
/usr/sbin/mknod c13t010 c 203 0x0d0000
/usr/sbin/mknod c13t014 c 203 0x0d0400
```

If you add the robots to NetBackup manually, you specify the following pathnames for robotic control. The first device file is for the HP EML E-Series robot. The second and third device files are for the VLS 6000 robot (two robotic devices).

```
/dev/sctl/c12t010
/dev/sctl/c13t010
/dev/sctl/c13t014
```

About creating legacy tape drive device files

By default, HP-UX creates tape drive device files when the system is booted. However, the tape driver must be installed and configured, and the devices must be attached and operational.

Alternatively, you can create tape drive device files manually. To do so, use either the HP-UX System Administration Manager (SAM) utility or the `insf(1M)` command. For information, see the HP-UX documentation.

Creating tape drive pass-through device files

On media servers, NetBackup creates pass-through paths for tape drives automatically. However, you can create them manually.

NetBackup also uses the tape drive pass-through device files for SAN Client.

Use one of the following two procedures:

- Create pass-through tape drive device files
See [“To create pass-through tape drive device files”](#) on page 44.

- Create SAN client pass-through device files
 See [“To create SAN client pass-through device files”](#) on page 46.

To create pass-through tape drive device files

- 1 Determine the devices that are attached to the SCSI bus by using the HP-UX `ioscan -f` command, as follows:

```
ioscan -f
Class      I  H/W Path      Driver S/W State  H/W Type  Description
=====
ext_bus    7  0/7/0/        c720  CLAIMED    INTERFACE  SCSI C896 Fast Wide LVD
target    10  0/7/0/1.0     tgt   CLAIMED    DEVICE      QUANTUM SuperDLT1
tape       65  0/7/0/1.0.0   stape CLAIMED    DEVICE      QUANTUM SuperDLT1
target    11  0/7/0/1.1     tgt   CLAIMED    DEVICE      QUANTUM SuperDLT1
tape       66  0/7/0/1.1.0   stape CLAIMED    DEVICE      QUANTUM SuperDLT1
target    12  0/7/0/1.2     tgt   CLAIMED    DEVICE      ADIC Scalar 100
autoch    14  0/7/0/1.2.0   schgr CLAIMED    DEVICE      IBM ULT3583-TL
target    13  0/7/0/1.3     tgt   CLAIMED    DEVICE      IBM ULT3580-TD1
autoch    19  0/7/0/1.3.0   schgr CLAIMED    DEVICE      IBM ULT3580-TD1
target    14  0/7/0/1.4     tgt   CLAIMED    DEVICE      IBM ULT3580-TD1
tape       21  0/7/0/1.4.0   atdd  CLAIMED    DEVICE      IBM ULT3580-TD1
target    15  0/7/0/1.5     tgt   CLAIMED    DEVICE      IBM ULT3580-TD1
tape       19  0/7/0/1.5.0   atdd  CLAIMED    DEVICE      IBM ULT3580-TD1
```

This example output shows the following:

- The robotic control for an ADIC Scalar 100 library is on a SCSI bus with an instance number of 7. The SCSI ID is 2, and the LUN is 0. The robotic control for an IBM ULT3583-TL library is on the same SCSI bus at SCSI ID 3 and LUN 0.
- The ADIC library contains two Quantum Super DLT drives. One has a SCSI ID of 0 and a LUN of 0. The other has a SCSI ID of 1 and a LUN of 0.
- The IBM library contains two IBM Ultrium LTO drives. One has a SCSI ID of 4 and a LUN of 0. The other has a SCSI ID of 5 and a LUN of 0. Use the IBM `atdd` driver when you configure IBM tape drives on HP-UX. Configure `atdd` and BEST device paths according to the IBM driver documentation. Do not configure `atdd` for robotic control of IBM robots. For

the latest recommended `atdd` driver version from IBM, check the Veritas support Web site.

2 Create the pass-through device files for the tape drives, as follows:

```
cd /dev/sctl  
/usr/sbin/mknod c7t010 c 203 0x070000  
/usr/sbin/mknod c7t110 c 203 0x071000  
/usr/sbin/mknod c7t410 c 203 0x074000  
/usr/sbin/mknod c7t510 c 203 0x075000
```

When you use the HP-UX `mknod` command for tape drives, the target is the SCSI ID of the tape drive. It is not the SCSI ID of the robotic control.

The previous commands create the following pass-through device files.

```
/dev/sctl/c7t010  
/dev/sctl/c7t110  
/dev/sctl/c7t410  
/dev/sctl/c7t510
```

Although the pass-through device files for tape drives are used during NetBackup operation, they are not used during NetBackup configuration. During NetBackup tape drive configuration, use the following device files to configure the tape drives.

```
/dev/rmt/c7t0d0BESTnb  
/dev/rmt/c7t1d0BESTnb  
/dev/rmt/c7t4d0BESTnb  
/dev/rmt/c7t5d0BESTnb
```

To create SAN client pass-through device files

- 1** Determine the devices that are attached to the SCSI bus by using the HP-UX `ioscan -f` command, as follows:

```
ioscan -f
Class      I  H/W Path                Driver      S/W State   H/W Type   Description
=====
ext_bus    9  0/3/1/0.1.22.255.0      fcd_vbus    CLAIMED     INTERFACE  FCP Device Interface
target     4  0/3/1/0.1.22.255.0.0    tgt         CLAIMED     DEVICE      ARCHIVE Python
tape       6  0/3/1/0.1.22.255.0.0.0  stape      CLAIMED     DEVICE      ARCHIVE Python
tape       7  0/3/1/0.1.22.255.0.0.1  stape      CLAIMED     DEVICE      ARCHIVE Python
```

This example output shows that the instance number of the Fibre Channel HBA is 9. It also shows that the target mode drivers on the Fibre Transport media server appear as `ARCHIVE Python` devices. One has a SCSI ID of 0 and a LUN of 0; the other has a SCSI ID of 0 and a LUN of 1.

- 2** Determine the character major number of the `sctl` driver by using the following command:

```
lsdev -d sctl
Character  Block  Driver  Class
203        -1     sctl    ctl
```

The output from this command shows that the character major number for the `sctl` driver is 203.

- 3** Create the pass-through device files, as follows:

```
cd /dev/sctl
/usr/sbin/mknod c9t010 c 203 0x090000
/usr/sbin/mknod c9t011 c 203 0x090100
```

The following describes the device file name:

- `c9` defines the instance number of the interface card.
- `t0` defines the SCSI ID (the target).
- `11` defines the LUN (the first character is the letter “l”).

- 4** Verify that the device files were created, as follows:

```
# ls -l /dev/sctl
total 0
crw-r--r--  1 root      sys          203 0x090000 Nov  1 13:19 c9t010
crw-r--r--  1 root      sys          203 0x090100 Nov  1 13:19 c9t011
```

About SPC-2 SCSI reserve on HP-UX

By default, NetBackup uses SPC-2 SCSI reserve and release for tape drive reservations in shared drive environments. However, you must disable SPC-2 SCSI reserve and release in HP-UX. If you do not, conflicts occur between the operating system and NetBackup. The NetBackup Shared Storage Option provides shared drive functionality in NetBackup.

Veritas recommends that you use the SAM utility to disable SPC-2 SCSI reserve and release.

As an alternative to SCSI reserve and release, you can use SCSI persistent reserve in NetBackup for shared tape drive reservations. If you use SCSI persistent reserve, you still must disable SPC-2 SCSI reserve and release in HP-UX.

For more information about NetBackup and SCSI reservations, see the following:

- The description of the **Enable SCSI Reserve Media** host property in the [NetBackup Administrator's Guide, Volume I](#).
- The "How NetBackup reserves drives" topic in the [NetBackup Administrator's Guide, Volume II](#).

Disabling SPC-2 SCSI reserve in HP-UX

Use the following procedure to disable SPC-2 SCSI reserve in HP-UX.

To disable SPC-2 SCSI reserve in HP-UX

- 1 Set the `st_ats_enabled` kernel parameter to 0 (zero).
- 2 Reboot the system.

About disabling the HP-UX EMS Tape Device Monitor for a SAN

You should configure the Tape Device Monitor (`dm_stape`) so it does not run on HP-UX hosts in a SAN configuration. The Tape Device Monitor is a component of the Event Monitoring System (EMS). The EMS service periodically polls the tape devices to monitor their conditions. When a server polls the devices while another server uses a tape device, backup operations may time out and fail.

You can avoid the situation as follows:

- To disable EMS completely, run the HP-UX Hardware Monitoring Request Manager and select **(K) kill (disable) Monitoring**.

Invoke the Hardware Monitoring Request Manager by using the `/etc/opt/resmon/sbin/monconfig` command.

- To configure EMS so it does not log any events or poll devices, set the `POLL_INTERVAL` value to 0 (zero). The `POLL_INTERVAL` parameter is in the following HP-UX configuration file:
`/var/stm/config/tools/monitor/dm_stape.cfg`
EMS runs but does not send any SCSI commands.

HP-UX command summary

The following is a summary of commands that may be useful when you configure and verify devices. For usage examples, see the procedures in this chapter.

- `ioscan -C class -f`
Shows the information about the physical interfaces. Numeric information is displayed in decimal.
`class` is the type of interface, as follows:
 - `tape` specifies tape drives.
 - `ext_bus` specifies SCSI controllers.
- `mknod /dev/spt/cCARDtTARGETlLUN c Major 0xIITL00`
Creates the device files for SCSI robotic or tape drive controls.
The following describe the device file names:
 - `CARD` is the card instance number of the adapter.
 - `TARGET` is the SCSI ID of the robotic control.
 - `LUN` is the SCSI logical unit number (LUN) of the robot.
 - `Major` is the character major number (from the `lsdev` command).
 - `II` are two hexadecimal digits that represent the card instance number.
 - `T` is a hexadecimal digit that represents the SCSI ID of robotic control.
 - `L` is a hexadecimal digit that represents the SCSI LUN of the robotic control.
- `lsdev -d driver`
Displays the information about the SCSI robotic control drivers.
- `mksf -C tape -H hw-path -b BEST -u -n`
Creates the device files for tape drives. The `hw-path` is the hardware path of the tape drive, as specified by the `ioscan` command.

Linux

This chapter includes the following topics:

- [Before you begin on Linux](#)
- [About the required Linux SCSI drivers](#)
- [Verifying the Linux drivers](#)
- [About configuring robot and drive control for Linux](#)
- [Verifying the device configuration on Linux](#)
- [About SAN clients on Linux](#)
- [About SCSI persistent bindings for Linux](#)
- [About Emulex HBAs](#)
- [Utilities to test SCSI devices](#)
- [Linux command summary](#)

Before you begin on Linux

Observe the following important points when you configure the operating system:

- Verify that NetBackup supports your server platform and devices. The Veritas support Web site contains server platform compatibility information. For the compatibility information, see the NetBackup compatibility lists :
<http://www.netbackup.com/compatibility>
- For SCSI controlled libraries, NetBackup issues SCSI commands to the robotic devices. For NetBackup to function correctly, the properly named device files must exist. Information about how to configure device files is available. See “[About configuring robot and drive control for Linux](#)” on page 51.

- Verify that a SCSI low-level driver is installed for each HBA in your system, as follows:
 - Follow the HBA vendor's installation guide to install or load the driver in the kernel.
 - Configure the kernel for SCSI tape support and SCSI generic support.
 - Probe all LUNs on each SCSI device and enable the SCSI low-level driver for the HBA.
 - Enable multi-LUN support for the kernel according to the Linux documentation.

For more information, refer to your HBA vendor documentation.

- Multipath configurations (multiple paths to robots and drives) are supported only with the following configurations:
 - Native path (`/dev/nstx`, `/dev/sgx`)
 - The `sysfs` file system that is mounted on `/sys`

After you configure the hardware, add the robots and the drives to NetBackup.

About the required Linux SCSI drivers

To use SCSI tape drives and robotic libraries, the following drivers must be configured in the kernel or loaded as modules:

- Linux SCSI generic (sg) driver. This driver allows pass-through commands to SCSI tape drives and control of robotic devices.

If you do not use a pass-through driver, performance suffers. NetBackup and its processes use the pass-through driver as follows:

- To scan drives
- For SCSI reservations
- For SCSI locate-block operations
- For SAN error recovery
- For Quantum SDLT performance optimization
- To collect robot and drive information
- To collect Tape Alert information from tape drives
- For WORM tape support
- For future features and enhancements

- SCSI tape (st) driver. This driver allows the use of SCSI tape drives.
- Standard SCSI driver.
- SCSI-adapter driver.

The standard Enterprise Linux releases have the `sg` and the `st` modules available for loading. The modules are loaded as needed. Also, you can load these modules if they are not in the kernel. Use the following commands:

```
/sbin/modprobe st  
/sbin/modprobe sg
```

About the st driver debug mode

You can enable debug mode for the `st` tape driver. Debug mode echoes each command and its result to the system log. For details, see the Linux documentation.

Verifying the Linux drivers

NetBackup requires specific Linux drivers.

See [“About the required Linux SCSI drivers”](#) on page 50.

You can use the `/sbin/lsmmod` command to verify that the `st` and the `sg` drivers are loaded in the kernel.

To verify that the drivers are installed and loaded in the kernel

- ◆ Invoke the `lsmmod` command as follows:

```
lsmmod  
Module           Size           Used by  
sg                14844          0  
st                24556          0
```

About configuring robot and drive control for Linux

NetBackup supports SCSI control and API control of robotic devices. SCSI control includes Fibre Channel Protocol (FCP), which is SCSI over Fibre Channel.

You must configure the control method, as follows:

- SCSI or Fibre Channel Protocol control.

NetBackup uses device files to configure control for SCSI tape devices, including robotic devices. (A robotic device in a library moves the media between storage slots and the drives in the library.)

See [“About the Linux robotic control device files”](#) on page 52.

See [“About the Linux tape drive device files”](#) on page 52.

- API control over a LAN.
 - See the "ADIC Automated Media Library (AML)" topic of this guide.
 - See the "IBM Automated Tape Library (ATL)" topic of this guide.
 - See the "Sun StorageTek ACSLS robots" topic of this guide.

About the Linux robotic control device files

For robotic devices, NetBackup uses the `/dev/sgx` device files, where `x` is a decimal number from 0 to 255. Linux should create the device files automatically. If the device files do not exist, see the Linux documentation for information about how to create them.

If you use device discovery, NetBackup looks for `/dev/sgx` robotic control device files. NetBackup discovers the robotic control device files (and hence the devices) automatically. Alternatively, if you add a robot manually in NetBackup, you must enter the pathname to the device file for that robotic device.

About the Linux tape drive device files

For tape drive device files, NetBackup uses the `/dev/nstx` files (`n` indicates the no rewind device file). The Linux driver should create the device files automatically. If the device files do not exist, see the Linux documentation for information about how to create them.

If you use device discovery in NetBackup, NetBackup looks for `/dev/nstx` device files. NetBackup discovers the device files (and hence the devices) automatically. Alternatively, if you add a drive manually in NetBackup, you must enter the pathname to the device file for that drive.

The NetBackup `avrd` daemon establishes a default tape driver operating mode. If you change the default mode, NetBackup may not read and write tapes correctly, which results in data loss.

Verifying the device configuration on Linux

The `/proc/scsi/scsi` file shows all of the devices that the SCSI driver detects.

If the operating system detects the SCSI devices, NetBackup can discover them.

To verify that the operating system can see the devices

- ◆ Run the following command from a terminal window:

```
cat /proc/scsi/scsi
```

The output that is displayed should be similar to the following:

```
Attached devices:
Host: scsi0 Channel: 00 Id: 01 Lun: 00
Vendor: HP          Model: C7200-8000      Rev: 1040
Type:   Medium Changer                    ANSI SCSI revision: 03
Host: scsi0 Channel: 00 Id: 02 Lun: 00
Vendor: QUANTUM    Model: DLT8000      Rev: 010F
Type:   Sequential-Access                  ANSI SCSI revision: 02
Host: scsi0 Channel: 00 Id: 03 Lun: 00
Vendor: QUANTUM    Model: DLT8000      Rev: 010F
Type:   Sequential-Access                  ANSI SCSI revision: 02
```

About SAN clients on Linux

NetBackup SAN clients on Linux hosts require the SCSI Generic (sg) driver and pass-through tape drive device files for traffic to NetBackup FT media servers. The media server FT devices appear as `ARCHIVE Python` tape devices during SCSI inquiry from the SAN client. (However, they are not tape devices and do not appear as tape devices in NetBackup device discovery.)

You should verify that you have the correct driver and device files.

See [“Verifying the Linux drivers”](#) on page 51.

If your Linux operating system does not add all of the SCSI device files automatically, you can do so manually. The following is an example of code you can include in the `/etc/rc.local` file to add LUN 1, targets 0-7 on Controllers 0-2. Note that the last line is the `MAKEDEV` command, which makes the required device files. The code you include in your `/etc/rc.local` file depends on how your hardware environment.

```
# Add the troublesome device on LUN 1 for the FT server
echo "scsi add-single-device 0 0 0 1" > /proc/scsi/scsi
echo "scsi add-single-device 0 0 1 1" > /proc/scsi/scsi
echo "scsi add-single-device 0 0 2 1" > /proc/scsi/scsi
echo "scsi add-single-device 0 0 3 1" > /proc/scsi/scsi
echo "scsi add-single-device 0 0 4 1" > /proc/scsi/scsi
echo "scsi add-single-device 0 0 5 1" > /proc/scsi/scsi
echo "scsi add-single-device 0 0 6 1" > /proc/scsi/scsi
echo "scsi add-single-device 0 0 7 1" > /proc/scsi/scsi
```

```
echo "scsi add-single-device 1 0 0 1" > /proc/scsi/scsi
echo "scsi add-single-device 1 0 1 1" > /proc/scsi/scsi
echo "scsi add-single-device 1 0 2 1" > /proc/scsi/scsi
echo "scsi add-single-device 1 0 3 1" > /proc/scsi/scsi
echo "scsi add-single-device 1 0 4 1" > /proc/scsi/scsi
echo "scsi add-single-device 1 0 5 1" > /proc/scsi/scsi
echo "scsi add-single-device 1 0 6 1" > /proc/scsi/scsi
echo "scsi add-single-device 1 0 7 1" > /proc/scsi/scsi
echo "scsi add-single-device 2 0 0 1" > /proc/scsi/scsi
echo "scsi add-single-device 2 0 1 1" > /proc/scsi/scsi
echo "scsi add-single-device 2 0 2 1" > /proc/scsi/scsi
echo "scsi add-single-device 2 0 3 1" > /proc/scsi/scsi
echo "scsi add-single-device 2 0 4 1" > /proc/scsi/scsi
echo "scsi add-single-device 2 0 5 1" > /proc/scsi/scsi
echo "scsi add-single-device 2 0 6 1" > /proc/scsi/scsi
echo "scsi add-single-device 2 0 7 1" > /proc/scsi/scsi
/dev/MAKEDEV sg
```

About SCSI persistent bindings for Linux

Veritas recommends that you use persistent bindings to lock the mappings between the SCSI targets that are reported to Linux and the specific devices. The Linux kernel device manager `udev` is often used to configure persistent bindings.

If you cannot use binding with the HBA in your configuration, add an `ENABLE_AUTO_PATH_CORRECTION` entry in the `/usr/opensv/volmgr/vm.conf` file on all Linux media servers.

About Emulex HBAs

If you use a `/usr/opensv/volmgr/AVRD_DEBUG` touch file on a NetBackup media server with an Emulex HBA driver, the system log may contain entries similar to the following:

```
Unknown drive error on DRIVENAME (device N, PATH) sense[0] = 0x70,
sense[1] = 0x0, sensekey = 0x5
```

You can ignore these messages.

Utilities to test SCSI devices

You can manipulate tape devices with the operating system `mt` command. For more information, see the `mt(1)` man page.

You can use the NetBackup `robtest` utility to test robots. The `robtest` utility resides in `/usr/opensv/volmgr/bin`.

A set of SCSI utilities are available from the Linux SCSI Generic (`sg`) driver home page.

Linux command summary

The following is a summary of commands that were used in this topic:

- `/sbin/lsmmod`
Lists the modules that are loaded.
- `/sbin/modprobe`
Installs loadable kernel modules.
- `/usr/sbin/reboot`
Stops and restarts the system.
- `/bin/mknod /dev/sgx c 21 N`
Creates SCSI generic device files; `x` is a decimal number from 0 to 255.

Solaris

This chapter includes the following topics:

- [Before you begin on Solaris](#)
- [About the NetBackup sg driver](#)
- [Determining if the NetBackup sg driver is installed](#)
- [Special configuration for the Oracle StorEdge Network Foundation HBA driver](#)
- [About binding Fibre Channel HBA drivers](#)
- [Configuring Solaris 10 x86 for multiple drive paths](#)
- [Installing/reinstalling the sg and the st drivers](#)
- [Configuring 6 GB and larger SAS HBAs in Solaris](#)
- [Preventing Solaris driver unloading](#)
- [About Solaris robotic controls](#)
- [About Solaris tape drive device files](#)
- [Configuring Solaris SAN clients to recognize FT media servers](#)
- [Uninstalling the sg driver on Solaris](#)
- [Solaris command summary](#)

Before you begin on Solaris

Observe the following points when you configure the operating system:

- Verify that NetBackup supports your server platform and devices. Download the NetBackup hardware and the operating system compatibility lists.

<http://www.netbackup.com/compatibility>

- For SCSI controlled libraries, NetBackup issues SCSI commands to the robotic devices.

For NetBackup to function correctly, the properly named device files must exist, as follows:

- NetBackup installs its own pass-through driver, the SCSI generic `sg` driver. You must configure this driver properly to create device files for any device NetBackup uses.
- The Solaris tape and disk driver interfaces also create a device file for each tape drive device. These device files must exist for all read or write I/O capability.

See “[About Solaris robotic controls](#)” on page 68.

- Verify that the Solaris `st` driver is installed.
- Verify that the devices are configured correctly. To do so, use the Solaris `mt` command and the NetBackup `/usr/opensv/volmgr/bin/sgscan` utility. For the tape drives that you want to share among NetBackup hosts, ensure that the operating system detects the devices on the SAN.
- When you configure devices, you should attach all peripherals and restart the system with the reconfigure option (`boot -r` or `reboot -- -r`).
- If you remove or replace adapter cards, remove all device files that are associated with that adapter card.
- If you use the Automated Cartridge System (ACS) robotic software, you must ensure that the Solaris Source Compatibility Package is installed. The package is required so that the ACS software can use the shared libraries in `/usr/ucb/lib`.
- Oracle systems with parallel SCSI host bus adapters do not support 16-byte SCSI commands on any devices that are attached to these HBAs. Therefore, those HBAs do not support WORM media. To override this limitation, create a touch file as follows:

```
touch /usr/opensv/volmgr/database/SIXTEEN_BYTE_CDB
```

After you configure the hardware, add the robots and the drives to NetBackup.

About the NetBackup sg driver

NetBackup provides its own SCSI pass-through driver to communicate with SCSI-controlled robotic peripherals. This driver is called the SCSA (generic SCSI pass-through driver), also referred to as the `sg` driver.

For full feature support, NetBackup requires the `sg` driver and SCSI pass-through device paths.

Install the NetBackup `sg` driver on each Solaris NetBackup media server that hosts tape devices. Each time you add or remove a device, you should reinstall the `sg` driver again.

If you do not use a pass-through driver, performance suffers.

NetBackup uses the pass-through driver for the following:

- By `avrd` and robotic processes to scan drives.
- By NetBackup to position tapes by using the locate-block method.
- By NetBackup for SAN error recovery.
- By NetBackup for Quantum SDLT performance optimization.
- By NetBackup for SCSI reservations.
- By NetBackup device configuration to collect robot and drive information.
- To collect Tape Alert information from tape devices allowing support of functions such as tape drive cleaning.
- For WORM tape support.
- Future NetBackup features and enhancements

Note: Because NetBackup uses its own pass-through driver, NetBackup does not support the Solaris `sgen` SCSI pass-through driver.

See [“Installing/reinstalling the sg and the st drivers”](#) on page 60.

Determining if the NetBackup sg driver is installed

Use the following procedure to determine if the `sg` driver is installed and loaded.

More information about the driver is available.

See [“About the NetBackup sg driver”](#) on page 57.

To determine if the sg driver is installed and loaded

- ◆ Invoke the following command:

```
/usr/sbin/modinfo | grep sg
```

If the driver is loaded, output includes a line similar to the following:

```
141 fc580000 2d8c 116 1 sg (SCSA Generic Revision: 3.5e)
```

Special configuration for the Oracle StorEdge Network Foundation HBA driver

When you configure the `sg` driver, it binds the StorEdge Network Foundation host bus adapter World Wide Port Names for use by the `sg` driver.

See [“Installing/reinstalling the `sg` and the `st` drivers”](#) on page 60.

The configuration process uses the Solaris `luxadm` command to probe for HBAs that are installed in the system. Ensure that the `luxadm` command is installed and in the shell path. For Solaris 11 and later, NetBackup uses the Solaris `sasinfo` command to probe for SAS attached devices.

To determine if a host contains a StorEdge Network Foundation HBA, you can run the following command:

```
/usr/opensv/volmgr/bin/sgscan
```

If the script detects a StorEdge Network Foundation HBA, it produces output similar to the following example:

```
#WARNING: detected StorEdge Network Foundation connected devices not
           in sg configuration file:
#
#   Device World Wide Port Name 21000090a50001c8
#
#   See /usr/opensv/volmgr/NetBackup_DeviceConfig_Guide.txt topic
#   "Special configuration for Sun StorEdge Network Foundation
#   HBA/Driver" for information on how to use sg.build and
#   sg.install to configure these devices
```

Each time you add or remove a device, you should configure the NetBackup `sg` driver and the Sun `st` driver again.

See [“About the NetBackup `sg` driver”](#) on page 57.

For 6 GB and larger serial attached SCSI (SAS) HBAs, also configure class 08 and 0101 for the `sg` driver.

See [“Configuring 6 GB and larger SAS HBAs in Solaris”](#) on page 65.

About binding Fibre Channel HBA drivers

For Fibre Channel HBAs other than Sun StorEdge Network Foundation, you must bind the devices to specific target IDs on the NetBackup host. When you bind devices to targets, the target ID does not change after a system reboot or a Fibre Channel configuration change.

In some instances, Veritas products are configured to use a specific target ID. If you change the ID, the products fail until you configure the ID correctly.

How you bind devices to targets is vendor and product specific. For information about how to modify the HBA configuration files to bind devices to targets, see the documentation for the HBA.

The binding may be based on the following:

- Fibre Channel World Wide Port Name (WWPN)
- World Wide Node Name (WWNN)
- The destination target ID and LUN

After you bind the devices to target IDs, continue with the Solaris configuration in the same manner as for parallel SCSI installations.

See [“Installing/reinstalling the sg and the st drivers”](#) on page 60.

Each time you add or remove a device, you must update the bindings and then configure the `sg` and the `st` drivers again.

Configuring Solaris 10 x86 for multiple drive paths

To use multiple paths to the same tape drive, NetBackup requires that Solaris Multiplexed I/O (MPxIO) be disabled. MPxIO is enabled by default on Solaris 10 x86 systems.

Use the following procedure to disable MPxIO.

To disable MPxIO

- 1 Use a text editor to open the following file:

```
/kernel/drv/fp.conf
```

- 2 Change the `mpxio-disable` value from **no** to **yes**. After the change, the line in the file should appear as follows:

```
mpxio-disable="yes"
```

- 3 Save the changes and exit from the text editor.

Installing/reinstalling the sg and the st drivers

You must install the NetBackup `sg` driver and the Sun `st` driver on each Solaris NetBackup media server that hosts tape devices.

Each time you add or remove a device, you should configure the NetBackup `sg` driver and the Sun `st` driver again. For 6 GB and larger serial-attached SCSI (SAS) HBAs, also configure class 08 and 0101 for the `sg` driver.

See [“Configuring 6 GB and larger SAS HBAs in Solaris”](#) on page 65.

Before you configure the `sg` and the `st` drivers, ensure that all devices are turned on and connected to the HBA.

See [“About the NetBackup `sg` driver”](#) on page 57.

The `sg.build` command uses the Solaris `sasinfo` command to probe for SAS attached device paths. This command is only available on Solaris 11 and later. On Solaris 10 and earlier, you must configure the `sg` driver manually.

To install and configure the `sg` and the `st` drivers

- 1 Invoke the following two commands to run the NetBackup `sg.build` script:

```
cd /usr/opensv/volmgr/bin/driver
/usr/opensv/volmgr/bin/sg.build all -mt target -ml lun
```

The following describes the options:

- The `all` option creates the following files and populates them with the appropriate entries:
 - `/usr/opensv/volmgr/bin/driver/st.conf`
See [“st.conf file example”](#) on page 63.
 - `/usr/opensv/volmgr/bin/driver/sg.conf`
See [“sg.conf file example”](#) on page 63.
 - `/usr/opensv/volmgr/bin/driver/sg.links`
See [“sg.links file example”](#) on page 64.
- The `-mt target` option and argument specify the maximum target ID that is in use on the SCSI bus (or bound to an FCP HBA). The maximum value is 126. By default, the SCSI initiator target ID of the adapter is 7, so the script does not create entries for target ID 7.

- The `-ml lun` option and argument specify the maximum number of LUNs that are in use on the SCSI bus (or by an FCP HBA). The maximum value is 255.
- 2 Replace the following seven entries in the `/kernel/drv/st.conf` file with all of the entries from the `/usr/opensv/volmgr/bin/driver/st.conf` file:

```
name="st" class="scsi" target=0 lun=0;
name="st" class="scsi" target=1 lun=0;
name="st" class="scsi" target=2 lun=0;
name="st" class="scsi" target=3 lun=0;
name="st" class="scsi" target=4 lun=0;
name="st" class="scsi" target=5 lun=0;
name="st" class="scsi" target=6 lun=0;
```

You should make a backup copy of the `/kernel/drv/st.conf` file before you modify it.

- 3 Reboot the system with the reconfigure option (`boot -r` or `reboot -- -r`). During the boot process, the system probes all targets in the `st.conf` file for devices. It should create device files for all of the devices it discovers.
- 4 Verify that Solaris created the device nodes for all the tape devices by using the following command:

```
ls -l /dev/rmt/*cbn
```

- 5 Install the new `sg` driver configuration by invoking the following two commands:

```
/usr/bin/rm -f /kernel/drv/sg.conf
/usr/opensv/volmgr/bin/driver/sg.install
```

The NetBackup `sg.install` script does the following:

- Installs and loads the `sg` driver.
 - Copies the `/usr/opensv/volmgr/bin/driver/sg.conf` file to `/kernel/drv/sg.conf`.
 - Creates the `/dev/sg` directory and nodes.
 - Appends the `/usr/opensv/volmgr/bin/driver/sg.links` file to the `/etc/devlink.tab` file.
- 6 Verify that the `<command>sg</command>` driver finds all of the robots and tape drives.

st.conf file example

The following `/usr/opensv/volmgr/bin/driver/st.conf` file example shows targets 0-15 and LUNs 0-7.

```
name="st" class="scsi" target=0 lun=0;
name="st" class="scsi" target=0 lun=1;
name="st" class="scsi" target=0 lun=2;
name="st" class="scsi" target=0 lun=3;
name="st" class="scsi" target=0 lun=4;
name="st" class="scsi" target=0 lun=5;
name="st" class="scsi" target=0 lun=6;
name="st" class="scsi" target=0 lun=7;
name="st" class="scsi" target=1 lun=0;
name="st" class="scsi" target=1 lun=1;
name="st" class="scsi" target=1 lun=2;
.
<entries omitted for brevity>
.
name="st" class="scsi" target=15 lun=5;
name="st" class="scsi" target=15 lun=6;
name="st" class="scsi" target=15 lun=7;
```

sg.conf file example

The following `/usr/opensv/volmgr/bin/driver/sg.conf` file example shows targets 0-15 and LUNs 0-8. It also includes target entries for three StorEdge Network Foundation HBA ports.

The `sg.build -mt` option does not affect FCP targets, but the `-ml` option does. The Solaris `luxadm` command detected three ports (identified by their World Wide Names). Therefore, the `sg.build` script created entries for LUNs 0 through 7 for those three ports.

```
name="sg" class="scsi" target=0 lun=0;
name="sg" class="scsi" target=0 lun=1;
name="sg" class="scsi" target=0 lun=2;
name="sg" class="scsi" target=0 lun=3;
name="sg" class="scsi" target=0 lun=4;
name="sg" class="scsi" target=0 lun=5;
name="sg" class="scsi" target=0 lun=6;
name="sg" class="scsi" target=0 lun=7;
name="sg" class="scsi" target=1 lun=0;
name="sg" class="scsi" target=1 lun=1;
```

```
name="sg" class="scsi" target=1 lun=2;
...
<entries omitted for brevity>
...
name="sg" class="scsi" target=15 lun=5;
name="sg" class="scsi" target=15 lun=6;
name="sg" class="scsi" target=15 lun=7;
name="sg" parent="fp" target=0 lun=0 fc-port-wwn="500104f0008d53c3";
name="sg" parent="fp" target=0 lun=1 fc-port-wwn="500104f0008d53c3";
name="sg" parent="fp" target=0 lun=0 fc-port-wwn="500104f0008d53c6";
name="sg" parent="fp" target=0 lun=1 fc-port-wwn="500104f0008d53c6";
name="sg" parent="fp" target=0 lun=0 fc-port-wwn="500104f0008d53c9";
name="sg" parent="fp" target=0 lun=1 fc-port-wwn="500104f0008d53c9";
name="sg" parent="fp" target=0 lun=0 fc-port-wwn="500104f0008d53cc";
name="sg" parent="fp" target=0 lun=1 fc-port-wwn="500104f0008d53cc";
name="sg" parent="fp" target=0 lun=0 fc-port-wwn="500104f0008d53b9";
name="sg" parent="fp" target=0 lun=1 fc-port-wwn="500104f0008d53b9";
name="sg" parent="fp" target=0 lun=0 fc-port-wwn="500104f0008d53c3";
name="sg" parent="fp" target=0 lun=1 fc-port-wwn="500104f0008d53c3";
name="sg" parent="fp" target=0 lun=0 fc-port-wwn="500104f0008d53c6";
name="sg" parent="fp" target=0 lun=1 fc-port-wwn="500104f0008d53c6";
name="sg" parent="fp" target=0 lun=0 fc-port-wwn="500104f0008d53c9";
name="sg" parent="fp" target=0 lun=1 fc-port-wwn="500104f0008d53c9";
name="sg" parent="fp" target=0 lun=0 fc-port-wwn="500104f0008d53cc";
name="sg" parent="fp" target=0 lun=1 fc-port-wwn="500104f0008d53cc";
name="sg" parent="fp" target=0 lun=0 fc-port-wwn="500104f0008d53b9";
name="sg" parent="fp" target=0 lun=1 fc-port-wwn="500104f0008d53b9";
```

sg.links file example

The following `/usr/opensv/volmgr/bin/driver/sg.links` file example shows targets 0-15 and LUNs 0-7. It also includes entries for three StorEdge Network Foundation HBA ports.

The `sg.build -mt` option does not affect FCP targets, but the `-ml` option does. The Solaris `luxadm` command detected three ports (identified by their World Wide Names). Therefore, the `sg.build` script created entries for LUNs 0 through 7 for those three ports.

The field separator between the `addr=x, y;` field and the `sg/` field is a tab. The `addr=` field uses hexadecimal notation, and the `sg/` field uses decimal values.

```
# begin SCSA Generic devlinks file - creates nodes in /dev/sg
type=ddi_pseudo;name=sg;addr=0,0;          sg/c\N0t010
```

```
type=ddi_pseudo;name=sg;addr=0,1;      sg/c\N0t011
type=ddi_pseudo;name=sg;addr=0,2;      sg/c\N0t012
type=ddi_pseudo;name=sg;addr=0,3;      sg/c\N0t013
type=ddi_pseudo;name=sg;addr=0,4;      sg/c\N0t014
type=ddi_pseudo;name=sg;addr=0,5;      sg/c\N0t015
type=ddi_pseudo;name=sg;addr=0,6;      sg/c\N0t016
type=ddi_pseudo;name=sg;addr=0,7;      sg/c\N0t017
type=ddi_pseudo;name=sg;addr=1,0;      sg/c\N0t110
type=ddi_pseudo;name=sg;addr=1,1;      sg/c\N0t111
...
<entries omitted for brevity>
...
type=ddi_pseudo;name=sg;addr=f,5;      sg/c\N0t1515
type=ddi_pseudo;name=sg;addr=f,6;      sg/c\N0t1516
type=ddi_pseudo;name=sg;addr=f,7;      sg/c\N0t1517
type=ddi_pseudo;name=sg;addr=w500104f0008d53c3,0;  sg/c\N0t\A110
type=ddi_pseudo;name=sg;addr=w500104f0008d53c3,1;  sg/c\N0t\A111
type=ddi_pseudo;name=sg;addr=w500104f0008d53c6,0;  sg/c\N0t\A110
type=ddi_pseudo;name=sg;addr=w500104f0008d53c6,1;  sg/c\N0t\A111
type=ddi_pseudo;name=sg;addr=w500104f0008d53c9,0;  sg/c\N0t\A110
type=ddi_pseudo;name=sg;addr=w500104f0008d53c9,1;  sg/c\N0t\A111
type=ddi_pseudo;name=sg;addr=w500104f0008d53c9,0;  sg/c\N0t\A110
type=ddi_pseudo;name=sg;addr=w500104f0008d53cc,0;  sg/c\N0t\A110
type=ddi_pseudo;name=sg;addr=w500104f0008d53cc,1;  sg/c\N0t\A111
type=ddi_pseudo;name=sg;addr=w500104f0008d53b9,0;  sg/c\N0t\A110
type=ddi_pseudo;name=sg;addr=w500104f0008d53b9,1;  sg/c\N0t\A111
type=ddi_pseudo;name=sg;addr=w500104f0008d53c3,0;  sg/c\N0t\A110
type=ddi_pseudo;name=sg;addr=w500104f0008d53c3,1;  sg/c\N0t\A111
type=ddi_pseudo;name=sg;addr=w500104f0008d53c6,0;  sg/c\N0t\A110
type=ddi_pseudo;name=sg;addr=w500104f0008d53c6,1;  sg/c\N0t\A111
type=ddi_pseudo;name=sg;addr=w500104f0008d53c9,0;  sg/c\N0t\A110
type=ddi_pseudo;name=sg;addr=w500104f0008d53c9,1;  sg/c\N0t\A111
type=ddi_pseudo;name=sg;addr=w500104f0008d53cc,0;  sg/c\N0t\A110
type=ddi_pseudo;name=sg;addr=w500104f0008d53cc,1;  sg/c\N0t\A111
type=ddi_pseudo;name=sg;addr=w500104f0008d53b9,0;  sg/c\N0t\A110
type=ddi_pseudo;name=sg;addr=w500104f0008d53b9,1;  sg/c\N0t\A111
# end SCSA devlinks
```

Configuring 6 GB and larger SAS HBAs in Solaris

Use the procedure in this topic to configure the NetBackup `sg` driver for Oracle 6 GB and larger SAS HBAs on Solaris.

A separate topic describes how to install the NetBackup `sg` and Sun `st` drivers.

See “Installing/reinstalling the sg and the st drivers” on page 60.

Note: Support for Solaris 6 GB serial-attached SCSI (SAS) HBAs for tape devices requires a specific Solaris patch level. Ensure that you install the required patches. For supported Solaris versions, see the Oracle Support website.

To configure 6 GB and larger SAS HBAs in Solaris

- 1 Verify that the 6 GB SAS tape device path exists by running the following command in a shell window:

```
ls -l /dev/rmt | grep cbn
```

6 GB SAS tape devices should have `iport@` in the name path. The following is an example of the output (the tape drive address is highlighted):

```
lcbn -> ../../devices/pci@400/pci@0/pci@9/LSI,sas@0/iport@8/tape@w500104f000ba856a,0:cbn
```

- 2 Edit the `/etc/devlink.tab` file

Include the following lines for every 6 GB SAS tape drive in the `/etc/devlink.tab` file. Replace `drive_address` with the tape drive address; see the output from step 1 for the tape drive addresses.

```
type=ddi_pseudo;name=sg;addr=wdrive_address,0,1;    sg/c\N0t\A110
type=ddi_pseudo;name=sg;addr=wdrive_address,1,1;    sg/c\N0t\A111
```

Include the following lines for every 6 GB SAS robotic library in the `/etc/devlink.tab` file. Replace `drive_address` with the tape drive address; see the output from step 1 for the tape drive address.

```
type=ddi_pseudo;name=medium-changer;addr=wdrive_address,0;    sg/c\N0t\A110
type=ddi_pseudo;name=medium-changer;addr=wdrive_address,1;    sg/c\N0t\A111
```

The following are example entries for the `devlink.tab` file:

```
# SCSI devlinks for SAS-2 drives:
type=ddi_pseudo;name=sg;addr=w500104f000ba856a,0,1;    sg/c\N0t\A110
type=ddi_pseudo;name=sg;addr=w500104f000ba856a,1,1;    sg/c\N0t\A111
# SCSI devlinks for SAS-2 libraries:
type=ddi_pseudo;name=medium-changer;addr=w500104f000ba856a,0;    sg/c\N0t\A110
type=ddi_pseudo;name=medium-changer;addr=w500104f000ba856a,1;    sg/c\N0t\A111
```

- 3 Verify that the `sg` driver SCSI classes are 08 and 0101 by running the following command:

```
grep sg /etc/driver_aliases
```

The following is an example of the output:

```
sg "scsiclass,0101"  
sg "scsiclass,08"
```

- 4 If the `sg` driver SCSI classes are not 08 and 0101, reinstall the `sg` driver by using the following commands:

```
rem_drv sg  
update_drv -d -i "scsiclass,08" sgen  
add_drv -m '* 0600 root root' -i "'scsiclass,0101" "scsiclass,08"' sg
```

- 5 Restart the host.
- 6 Verify that the `sg` drivers exist by running the following command:

```
ls -l /dev/sg
```

The following is an example of the output (the output was modified to fit on the page):

```
c0tw500104f000ba856a10 ->  
../devices/pci@400/pci@0/pci@9/LSI,sas@0/iport@8/sg@w500104f000ba856a,0,1:raw  
c0tw500104f000ba856a11 ->  
../devices/pci@400/pci@0/pci@9/LSI,sas@0/iport@8/medium-changer@w500104f000ba856a,1:raw
```

- 7 Verify that the NetBackup `sgscan` utility recognizes the tape devices by entering the following command:

```
/usr/opensv/volmgr/bin/sgscan
```

The following is an example of the output:

```
/dev/sg/c0tw500104f000ba856a10: Tape (/dev/rmt/1): "HP      Ultrium 5-SCSI"  
/dev/sg/c0tw500104f000ba856a11: Changer: "STK      SL500"
```

Preventing Solaris driver unloading

When system memory is limited, Solaris unloads unused drivers from memory and reloads drivers as needed. Tape drivers are often unloaded because they are used less often than disk drivers.

The drivers NetBackup uses are the `st` driver (from Sun), the `sg` driver (from Veritas), and Fibre Channel drivers. Problems may occur depending on when the driver loads and unloads. These problems can range from a SCSI bus not able to detect a device to system panics.

Veritas recommends that you prevent Solaris from unloading the drivers from memory.

The following procedures describe how to prevent Solaris from unloading the drivers from memory.

To prevent Solaris from unloading the drivers from memory

- ◆ Add the following `forceload` statements to the `/etc/system` file:

```
forceload: drv/st
forceload: drv/sg
```

To prevent Solaris from unloading the Fibre Channel drivers from memory

- ◆ Add an appropriate `forceload` statement to the `/etc/system` file.

Which driver you force to load depends on your Fibre Channel adapter. The following is an example for a Sun Fibre Channel driver (SunFC FCP v20100509-1.143):

```
forceload: drv/fcp
```

About Solaris robotic controls

NetBackup supports SCSI control and API control of robotic devices. A robotic device in a library moves the media between the storage slots and the drives in the library.

Robotic control varies, as follows:

- SCSI or Fibre Channel Protocol control.
See [“About SCSI and FCP robotic controls on Solaris”](#) on page 68.
- API control over a LAN.
See the "ADIC Automated Media Library (AML)" topic in this guide.
See the "IBM Automated Tape Library (ATL)" topic in this guide.
See the "Sun StorageTek ACSLS robots" topic in this guide.

About SCSI and FCP robotic controls on Solaris

When you configure the NetBackup `sg` driver, a NetBackup script creates the device files for the attached robotic devices.

See [“About the NetBackup sg driver”](#) on page 57.

If you use device discovery in NetBackup, NetBackup discovers the robotic control device files in the `/dev/sg` directory (and hence the devices) automatically. If you add a robot manually in NetBackup, you must enter the pathname to the device file.

To display the device files that the `sg` driver can use, use the NetBackup `sgscan` command with the `all` parameter. The word "Changer" in the `sgscan` output identifies robotic control device files.

Examples are available.

See [“Examples of SCSI and FCP robotic control device files on Solaris”](#) on page 69.

Examples of SCSI and FCP robotic control device files on Solaris

The following is an example of `sgscan all` output from a host, to which the examples refer:

```
# /usr/opensv/volmgr/bin/sgscan all
/dev/sg/c0t6l0: Cdrom: "TOSHIBA XM-5401TASUN4XCD"
/dev/sg/cltw500104f0008d53b9l0: Changer: "STK      SL500"
/dev/sg/cltw500104f0008d53c3l0: Tape (/dev/rmt/0): "HP      Ultrium 3-SCSI"
/dev/sg/cltw500104f0008d53c6l0: Tape (/dev/rmt/1): "HP      Ultrium 3-SCSI"
/dev/sg/cltw500104f0008d53c9l0: Tape (/dev/rmt/2): "IBM     ULTRIUM-TD3"
/dev/sg/cltw500104f0008d53cc10: Tape (/dev/rmt/3): "IBM     ULTRIUM-TD3"
/dev/sg/c2t1l0: Changer: "STK      SL500"
/dev/sg/c2t2l0: Tape (/dev/rmt/22): "HP      Ultrium 3-SCSI"
/dev/sg/c2t3l0: Tape (/dev/rmt/10): "HP      Ultrium 3-SCSI"
/dev/sg/c2tal0: Tape (/dev/rmt/18): "IBM     ULTRIUM-TD3"
/dev/sg/c2tbl0: Tape (/dev/rmt/19): "IBM     ULTRIUM-TD3"
/dev/sg/c3t0l0: Disk (/dev/rdisk/c1t0d0): "FUJITSU MAV2073RCSUN72G"
/dev/sg/c3t3l0: Disk (/dev/rdisk/c1t3d0): "FUJITSU MAV2073RCSUN72G"
```

You can filter the `sgscan` output for device types by using other `sgscan` options. The following is the `sgscan` usage statement:

```
sgscan [all|basic|changer|disk|tape] [conf] [-v]
```

About Solaris tape drive device files

NetBackup uses the tape drive device files that support compression, no rewind on close, and Berkeley style close.

When you configure the Solaris `st` driver, Solaris creates the device files for the attached tape devices

See “Installing/reinstalling the `sg` and the `st` drivers” on page 60.

The device files are in the `/dev/rmt` directory, and they have the following format:

```
/dev/rmt/IDcn
```

The following describe the device file names:

- `ID` is the logical drive number as shown by the NetBackup `sgscan` command.
- `c` indicates compression.
- `b` indicates Berkeley-style close.
- `n` indicates no rewind on close.

If you use device discovery in NetBackup, NetBackup discovers the device files and hence the devices. If you add a tape drive to a NetBackup configuration manually, you must specify the pathname to the device file. NetBackup requires compression, no rewind on close, and Berkeley-style close device files.

To display the tape device files that are configured on your system, use the `sgscan` command with the `tape` parameter, as follows:

```
# /usr/opensv/volmgr/bin/sgscan tape
/dev/sg/cltw500104f0008d53c310: Tape (/dev/rmt/0): "HP      Ultrium 3-SCSI"
/dev/sg/cltw500104f0008d53c610: Tape (/dev/rmt/1): "HP      Ultrium 3-SCSI"
/dev/sg/cltw500104f0008d53c910: Tape (/dev/rmt/2): "IBM     ULTRIUM-TD3"
/dev/sg/cltw500104f0008d53cc10: Tape (/dev/rmt/3): "IBM     ULTRIUM-TD3"
/dev/sg/c2t210: Tape (/dev/rmt/22): "HP      Ultrium 3-SCSI"
/dev/sg/c2t310: Tape (/dev/rmt/10): "HP      Ultrium 3-SCSI"
/dev/sg/c2ta10: Tape (/dev/rmt/18): "IBM     ULTRIUM-TD3"
/dev/sg/c2tb10: Tape (/dev/rmt/19): "IBM     ULTRIUM-TD3"
```

The following are examples of no-rewind, compression, Berkeley-style close device files from the preceding `sgscan` example output:

- For the Ultrium3 SCSI drive at LUN 0 of World Wide Node Name (WWNN) 500104f0008d53c3, the device file pathname is:
`/dev/rmt/0cn`
- For the HP Ultrium3 SCSI drive at SCSI ID 2 of adapter 2, the device file pathname is:
`/dev/rmt/22cn`

You can show all device types by using the `all` option. The output can help you associate tape devices with other SCSI devices that may be configured on the same adapter. The following is the `sgscan` usage statement:

```
sgscan [all|basic|changer|disk|tape] [conf] [-v]
```

About Berkeley-style close

NetBackup requires Berkeley-style close for tape drive device files. The letter `b` in the file name indicates Berkeley-style close device files.

In Berkeley-style close, the tape position remains unchanged by a device close operation. (Conversely, in AT&T-style close, the drive advances the tape to just after the next end-of-file (EOF) marker.) To establish the correct position for the next tape operation, applications must assume the tape's position after a close. NetBackup assumes Berkeley-style close on Solaris systems.

About no rewind device files on Solaris

NetBackup requires no rewind on close device files for tape drives.

With no rewind on close, the tape is not rewound after a close operation. It remains in position for the next write operation.

The letter `n` in the device file names in the `/dev/rmt` directory specifies no rewind on close.

About fast-tape positioning (locate-block) on Solaris

Applies to AIT, DLT, Exabyte, DTF, and half-inch tape drives.

To position a tape to a specific block, NetBackup supports the SCSI `locate-block` command. It requires the NetBackup `sg` driver.

NetBackup uses the `locate-block` command by default.

Veritas recommends that you do not disable `locate-block` positioning. If you need to disable it, execute the following command:

```
touch /usr/openv/volmgr/database/NO_LOCATEBLOCK
```

If `locate-block` positioning is disabled, NetBackup uses the forward-space-file/record method.

About SPC-2 SCSI reserve on Solaris

By default, NetBackup uses SPC-2 SCSI reserve and release for tape drive reservations in shared drive environments. The NetBackup Shared Storage Option provides shared drive functionality in NetBackup.

Alternatively, you can use SCSI persistent reserve for shared tape drive reservations in NetBackup, as follows:

- For the tape drives that support SPC-3 Compatible Reservation Handling (CRH), you can use SCSI persistent reserve by enabling it in NetBackup. No special configuration in Solaris is required.
- For the tape drives that do not support CRH, you must disable SPC-2 SCSI reserve in Solaris for those drives. After you disable SPC-2 SCSI reserve, you can use persistent reserve by enabling it in NetBackup. If the drive does not support CRH and you do not disable SPC-2 SCSI reserve, access attempts to the drive fail.

See [“Disabling SPC-2 SCSI reserve on Solaris”](#) on page 72.

For more information about NetBackup and SCSI reservations, see the following:

- The description of the **Enable SCSI Reserve Media** host property in the [NetBackup Administrator’s Guide, Volume I](#).
- The “How NetBackup reserves drives” topic in the [NetBackup Administrator’s Guide, Volume II](#).

Disabling SPC-2 SCSI reserve on Solaris

Use the following procedure to disable SPC-2 SCSI reserve.

More information about reservations is available.

See [“About SPC-2 SCSI reserve on Solaris”](#) on page 72.

To disable SPC-2 SCSI reserve

- ◆ Modify the Solaris `st.conf` file on the NetBackup media server. In the `tape-config-list` section of the `st.conf` file, set the `ST_NO_RESERVE_RELEASE` configuration value (0x20000) in the appropriate `data-property-name` entry.

For example, the following entry disables SCSI reserve and release for all tape drives that use the `DLT7k-data` configuration values:

```
DLT7k-data = 1,0x38,0,0x20000,4,0x82,0x83,0x84,0x85,2;
```

For more information about the `st.conf` file, see the Solaris `st(7D)` man page.

About nonstandard tape drives

Solaris includes the device drivers that support most standard devices.

To receive the most current support for devices, you should install the latest Solaris patch for the `st` driver.

However, if you have a device that Solaris does not support, the device manufacturer should provide the software to install and administer the device properly. In addition, the device vendor should contact Sun Microsystems to add support for the device to Solaris.

For more information about what you need for unsupported devices, contact the device vendor. Also see the Solaris devices and file systems documentation.

Configuring Solaris SAN clients to recognize FT media servers

NetBackup SAN clients use tape drivers and SCSI pass-through methods for Fibre Transport traffic to NetBackup FT media servers. The media server FT devices appear as `ARCHIVE Python` tape devices during SCSI inquiry on the SAN client. However, they are not tape devices and do not appear as tape devices in NetBackup device discovery.

Veritas owns the `ARCHIVE` brand name and `Python` product name. Therefore, `st.conf` file changes to `ARCHIVE Python` do not affect an existing tape drive product.

[Table 5-1](#) is an overview of procedures to configure the Solaris operating system so that it recognizes the NetBackup FT devices on the NetBackup media servers.

Table 5-1 Configuring SAN clients to recognize FT media servers

Step	Task	Procedure
1	Add the Fibre Transport device entry to the <code>st.conf</code> file	See “Adding the FT device entry to the st.conf file” on page 73.
2	Modify the <code>st.conf</code> file so that Solaris discovers devices on two LUNS	See “Modifying the st.conf file so that Solaris discovers devices on two LUNS” on page 74.

Adding the FT device entry to the st.conf file

The following procedure describes how to add the FT device entry to the `st.conf` file.

To add the FT device entry to the st.conf file

- 1** In the `/kernel/drv/st.conf` file, find the `tape-config-list=` section or create it if it does not exist.
- 2** Examine the `tape-config-list=` section for a line that begins with `ARCHIVE Python` and contains `ARCH_04106`. If such a line exists, ensure that it begins with a comment character (`#`).

- 3** Add the following line to the `tape-config-list=` section:

```
"ARCHIVE Python", "FT Pipe", "ARCH_04106";
```

- 4** Find the line that begins with `ARCH_04106`, copy it, and paste it after the `tape-config-list=` line. Delete the comment character (`#`) from the beginning of the line. The following is an example of the line:

```
ARCH_04106 = 1, 0x2C, 0, 0x09639, 4, 0x00, 0x8C, 0x8c, 0x8C, 3;
```

Modifying the st.conf file so that Solaris discovers devices on two LUNS

The following procedure describes how to modify the `st.conf` file so that Solaris discovers devices on two LUNS.

To modify the st.conf file so that Solaris discovers devices on two LUNS

- 1 Find the following line in the `st.conf` file:

```
name="st" class="scsi" target=0 lun=0;
```

- 2 Replace that line and the following lines through target 5 with the following. Doing so modifies the `st.conf` file to include searches on non-zero LUNs.

```
name="st" class="scsi" target=0 lun=0;
name="st" class="scsi" target=0 lun=1;
name="st" class="scsi" target=1 lun=0;
name="st" class="scsi" target=1 lun=1;
name="st" class="scsi" target=2 lun=0;
name="st" class="scsi" target=2 lun=1;
name="st" class="scsi" target=3 lun=0;
name="st" class="scsi" target=3 lun=1;
name="st" class="scsi" target=4 lun=0;
name="st" class="scsi" target=4 lun=1;
name="st" class="scsi" target=5 lun=0;
name="st" class="scsi" target=5 lun=1;
name="st" parent="fp" target=0;
name="st" parent="fp" target=1;
name="st" parent="fp" target=2;
name="st" parent="fp" target=3;
name="st" parent="fp" target=4;
name="st" parent="fp" target=5;
name="st" parent="fp" target=6;
```

Uninstalling the sg driver on Solaris

You can uninstall the `sg` driver. If you do, NetBackup performance suffers. The following procedure describes how to uninstall the `sg` driver.

To uninstall the sg driver

- ◆ Invoke the following command:

```
/usr/sbin/rem_drv sg
```

Solaris command summary

The following is a summary of commands that may be useful when you configure and verify devices:

- `/usr/sbin/modinfo | grep sg`
Displays whether or not the `sg` driver is installed.
- `/usr/opensv/volmgr/bin/driver/sg.install`
Installs the `sg` driver or updates the `sg` driver.
- `/usr/sbin/rem_drv sg`
Uninstalls the `sg` driver. This command usually is not necessary because `sg.install` uninstalls the old driver before it upgrades a driver.
- `/usr/opensv/volmgr/bin/sg.build all -mt max_target -ml max_lun`
Updates `st.conf`, `sg.conf`, and `sg.links`, and generates SCSI Target IDs with multiple LUNs.
- `/usr/opensv/volmgr/bin/sgscan all`
Scans all connected devices with an SCSI inquiry and provides correlation between physical and the logical devices that use all device files in `/dev/sg`. Also checks for the devices that are connected to the Sun StorEdge Network Foundation HBA that are not configured for use by Veritas products.
- `boot -r` or `reboot -- -r`
Reboot the system with the reconfigure option (`-r`). The kernel's SCSI disk (`sd`) driver then recognizes the drive as a disk drive during system initialization.

See the procedures in this chapter for examples of their usage.

Windows

This chapter includes the following topics:

- [Before you begin configuring NetBackup on Windows](#)
- [About tape device drivers on Windows](#)
- [Attaching devices to a Windows system](#)

Before you begin configuring NetBackup on Windows

Observe the following points when performing the configurations described in this chapter:

- Verify that NetBackup supports your server platform and devices. Download the NetBackup hardware and operating system compatibility lists:
<http://www.netbackup.com/compatibility>
- For NetBackup to recognize and communicate with connected devices and for device discovery to discover devices, NetBackup issues SCSI pass-through commands to the devices in a configuration.
A tape driver must exist for each tape device. Attached devices appear in the registry.
- Use the Microsoft Windows device applications to verify that the devices are configured correctly. The device applications available on your server may differ depending on your Windows operating system. Make sure that Windows detects the devices on the SAN before you configure the NetBackup Shared Storage Option.
- If you have multiple devices connected to a fibre bridge, Windows may only see one LUN. This will normally be the device with the lowest-ordered LUN.

This limitation occurs because of the default install settings for the device driver for some fibre channel HBAs. See your vendor documentation to verify the settings.

- Information about how to configure API robot control over a LAN is available. See the "ADIC Automated Media Library (AML)" topic in this guide. See the "IBM Automated Tape Library (ATL)" topic in this guide. See the "Sun StorageTek ACSLS robots" topic in this guide.

After configuring the hardware, add the drives and robots to NetBackup.

About tape device drivers on Windows

Veritas does not provide device drivers for Windows hosts. If you require drivers, contact Microsoft or the tape drive vendor.

Attaching devices to a Windows system

The following procedure describes a general method for attaching devices to a Windows computer. The Microsoft Windows device applications available on the server that you use in these steps may differ depending on your Windows operating system.

To attach devices to a Windows system

- 1 Use the appropriate Windows application to obtain information on any currently attached SCSI devices.
- 2 If you attach a new robotic library or drive to a NetBackup media server, follow the vendor's instructions for attaching the device.

Shut down the server and physically attach the supported device. Ensure that SCSI targets and termination settings are consistent with adapter card and peripheral vendor recommendations.
- 3 Reboot the server and answer the prompts for adapter card peripheral configuration options. Watch the display to ensure that the adapter card recognizes the attached peripherals.
- 4 If you add drives, install the tape drivers and use the appropriate Windows application to verify that the drive was recognized.

Robotic storage devices

- [Chapter 7. Robot overview](#)
- [Chapter 8. Oracle StorageTek ACSLS robots](#)
- [Chapter 9. Device configuration examples](#)

Robot overview

This chapter includes the following topics:

- [NetBackup robot types](#)
- [NetBackup robot attributes](#)
- [Table-driven robotics](#)
- [Robotic test utilities](#)
- [Robotic processes](#)

NetBackup robot types

A robot is a peripheral device that moves tape volumes into and out of tape drives. NetBackup uses robotic control software to communicate with the robot firmware.

NetBackup classifies robots according to one or more of the following characteristics:

- The communication method the robotic control software uses; SCSI and API are the two main methods.
- The physical characteristics of the robot. Library refers to a large robot, in terms of slot capacity or number of drives.
- The media type commonly used by that class of robots. HCART (1/2-inch cartridge tape) is an example of a media type.

[Table 7-1](#) lists the NetBackup robot types that are supported in release 8.1, with drive and slot limits for each type.

To determine which robot type applies to the model of robot that you use, see the hardware compatibility list for your NetBackup version:

<http://www.netbackup.com/compatibility>

Table 7-1 NetBackup robot types in release 8.1

Robot type	Description	Drive limits	Slot limits	Note
ACS	Automated Cartridge System	1680	No limit	API control. The ACS library software host determines the drive limit.
TLD	Tape library DLT	No limit	32000	SCSI control.

Note: The user interface for NetBackup may show configuration options for the peripheral devices that are not supported in that release. Those devices may be supported in an earlier release, and a NetBackup master server can manage the hosts that run earlier NetBackup versions. Therefore, the configuration information for such devices must appear in the user interface. The NetBackup documentation also may describe the configuration information for such devices. To determine which versions of NetBackup support which peripheral devices, see the hardware compatibility list:

<http://www.netbackup.com/compatibility>

For information about the robots that NetBackup supports, see the hardware compatibility for your release available through the following URL:

<http://www.netbackup.com/compatibility>

NetBackup robot attributes

NetBackup configures and controls robots differently depending on the robot type. The following tables list the attributes that dictate how these robot types differ.

For more detailed information about supported devices, firmware levels, and platforms, see the hardware compatibility list for your NetBackup version:

<http://www.netbackup.com/compatibility>

See “NetBackup robot types” on page 80.

ACS robots

Unlike other robot types, NetBackup does not track slot locations for the media in ACS robots. The ACS library software tracks slot locations and reports them to NetBackup.

The following table describes the ACS robot attributes.

Table 7-2 ACS robot attributes

Attribute	NetBackup server
API robot	Yes
SCSI control	No
LAN control	Yes
Remote Robot control	No. Each host that has ACS drives that are attached to it has robotic control.
NDMP support	Yes
Shared drives support	Yes
Drive cleaning support	No. The ACS library software manages drive cleaning.
Media access port support	Yes, for eject only.
NetBackup tracks slots	No
Media type support	DLT, DLT2, DLT3, HCART, HCART2, and HCART3.
Hosts Supported	Windows, UNIX, and Linux. Windows servers require STK LibAttach software. See the Veritas support web site for the latest compatibility information and obtain the appropriate LibAttach software from STK.
Barcode Support	Yes. Depends on ACS library software to obtain NetBackup media IDs. Barcodes must be the same as the media ID (1 to 6 characters).
Robot Examples	Oracle SL500, Oracle SL3000, and Oracle SL8500

TLD robots

The following table describes the tape library DLT attributes.

Table 7-3 TLD robot attributes

Attribute	NetBackup Server	NetBackup server
API robot	No	No
SCSI control	Yes	Yes

Table 7-3 TLD robot attributes (*continued*)

Attribute	NetBackup Server	NetBackup server
LAN control	Not Applicable	No
Remote robot control	Not Applicable	Yes
NDMP support	Yes	Yes
Shared drives support	Not Applicable	Yes
Drive cleaning support	Yes	Yes
Media access port support	Yes	Yes
NetBackup tracks slots	Yes	Yes
Hosts supported	Windows, UNIX, and Linux.	Windows, UNIX, and Linux.
Media type support	DLT, DLT2, DLT3, DTF, 8MM, 8MM2, 8MM3, QIC, HCART, HCART2, HCART3	DLT, DLT2, DLT3, DTF, 8MM, 8MM2, 8MM3, QIC, HCART, HCART2, HCART3
Barcode support	Yes. Barcodes can be from 1 to 16 characters in length. The Media Manager media ID is six or fewer characters.	Yes. Barcodes can be from 1 to 16 characters in length. The Media Manager media ID is six or fewer characters.
Robot examples	HP MSL, Fujitsu FibreCAT TX48, IBM TotalStorage3583, Spectra Logic T680, Sun/Oracle SL3000	HP MSL, Fujitsu FibreCAT TX48, IBM TotalStorage3583, Spectra Logic T680, Sun/Oracle SL3000

Table-driven robotics

Table-driven robotics provides support for new robotic library devices without the need to modify any library control binary files. This feature uses a device mapping file for supported robots and drives.

You may be able to add support for new or upgraded devices without waiting for a maintenance patch from Veritas. The device mapping file includes the information that relates to the operation and control of libraries. Therefore, you can download an updated mapping file to obtain support for newly NetBackup-certified devices.

For the device mappings file downloads, see the following URL:

<http://www.netbackup.com/compatibility>

See “[NetBackup robot types](#)” on page 80.

Robotic test utilities

You can use robotic test utilities for testing robots already configured in NetBackup.

Invoke the test utilities as follows:

- `/usr/opensv/volmgr/bin/robtest` (UNIX and Linux)
- `install_path\Veritas\Volmgr\bin\robtest.exe` (Windows)

From each test utility, you can obtain a list of available test commands by entering a question mark (?).

Use the `drstat` command to determine the drive addressing parameters for the ACS robot type. This command is available in the robotic test utilities for these robot types.

NetBackup addresses drives as follows:

- For ACS robot types, by ACS, LSM, Panel, and Drive number
- For other robot types, by the robot drive number

See “[NetBackup robot types](#)” on page 80.

Robotic processes

A NetBackup robotic process and possibly a robotic control process exist on a NetBackup media server for each robot that you install, as follows:

- Every media server that has a drive in a robotic library has a robotic process for that robotic library. The robotic process receives requests from the NetBackup Device Manager (`ltid`) and sends necessary information directly to the robotics or to a robotic control process.
- Robotic control processes exist only for the robot types that support library sharing (or robot sharing).

When the NetBackup Device Manager starts, it starts the robotic processes and the robotic control processes for all of the configured robots on that host. When the Device Manager stops, the robotic processes and the robotic control processes stop. (On UNIX, the name is Media Manager Device daemon.)

You can start and stop the Device Manager manually by using the **NetBackup Administration Console** as follows:

- In the NetBackup **Activity Monitor Daemons** tab, select it and then select **Actions Start Daemon** or **Stop Daemon**.

- Select **Device Monitor**, **Media**, or **Devices** in the right pane and then select **Actions Stop/Restart Media Manager Device Daemon**.

In addition, the [NetBackup Commands Reference Guide](#) describes commands to control the robotic processes that run on Windows media servers.

You can determine if a robotic process or robotic control process is active by using the NetBackup **Activity Monitor Processes** tab.

You can determine the control state of NetBackup by using the **Device Monitor Drive paths** pane or **Drive status** pane. If the value in the **Control** column for a drive shows the control mode, the robotic process is running and the drive is usable. For example, for a TLD robot the control mode is **TLD**.

Other values such as **AVR** or **DOWN** may indicate that the drive is unusable. For the possible values and their explanations, see the online Help for the **Device Monitor**.

See [“Processes by robot type”](#) on page 85.

See [“Robotic process example”](#) on page 86.

See [“NetBackup robot types”](#) on page 80.

Processes by robot type

The following table describes the robotic processes and robotic control processes for each robot type.

Table 7-4 Robotic processes and robotic control processes

Robot type	Process	Description
Automated Cartridge System (ACS)	acsd	The NetBackup ACS daemon <code>acsd</code> provides robotic control to mount and dismount volumes. It also requests inventories of the volumes that are under the control of ACS library software.
	acssel	The NetBackup ACS storage server interface (SSI) event logger <code>acssel</code> logs events. UNIX and Linux only.
	acsssi	The NetBackup ACS storage server interface (SSI) <code>acsssi</code> communicates with the ACS library software host. <code>acsssi</code> processes all RPC communications from <code>acsd</code> or from the ACS robotic test utility that are intended for the ACS library software. UNIX and Linux only.
Tape library DLT (TLD)	tidd	The tape library DLT daemon <code>tidd</code> runs on a NetBackup server that has a drive in the tape library DLT. This process receives NetBackup Device Manager requests to mount and unmount volumes, and sends these requests to the robotic-control process, <code>tldcd</code> .

Table 7-4 Robotic processes and robotic control processes (*continued*)

Robot type	Process	Description
	tlbcd	The tape library DLT Control daemon <code>tlbcd</code> communicates with the tape library DLT robotics through a SCSI interface. For library sharing, <code>tlbcd</code> runs on the NetBackup server that has the robotic control.
Tape library 4MM (TL4)	tl4d	The tape library 4MM daemon <code>tl4d</code> runs on the host that has a tape library 4MM. This process receives NetBackup Device Manager requests to mount and unmount volumes and communicates these requests to the robotics through a SCSI interface.
Tape library 8MM (TL8)	tl8d	The tape library 8MM daemon <code>tl8d</code> runs on a NetBackup server that has a drive in the tape library 8MM. This process receives NetBackup Device Manager requests to mount and unmount volumes, and sends these requests to the robotic-control process, <code>tl8cd</code> .
	tl8cd	The tape library 8MM Control daemon <code>tl8cd</code> communicates with the TL8 robotics through a SCSI interface. For library sharing, <code>tl8cd</code> runs on the NetBackup server that has the robotic control.
Tape library Half-inch (TLH)	tlhd	The tape library Half-inch daemon <code>tlhd</code> runs on each NetBackup server that has a drive in the tape library Half-inch. This process receives NetBackup Device Manager requests to mount and unmount volumes and sends these requests to the robotic-control process.
	tlhcd	The tape library half-inch control daemon <code>tlhcd</code> runs on the NetBackup server that has the robotic control. It communicates with the TLH robotics through a SCSI interface
Tape library Multimedia (TLM)	tlmd	<code>tlmd</code> runs on the NetBackup server and communicates mount and unmount requests to the host that controls the TLM robotics.

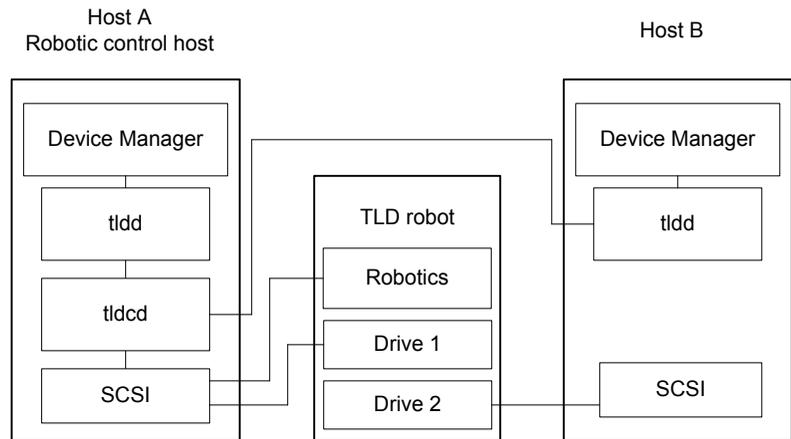
See “[NetBackup robot types](#)” on page 80.

Robotic process example

Each drive in a tape library DLT (TLD) robot can be attached to a different host, and a `tldd` process runs on each host. However, only one host controls the robotics, and the `tlbcd` robotic control process runs on that host only. To mount a tape, the `tldd` process on the host to which the drive is attached sends control information to the `tlbcd` process on the robotic control host.

The following figure shows the processes and where they run for a TLD robot.

Figure 7-1 TLD robot control process example



The following describes this example:

- Each host connects to one drive, and a `t1dd` robotic process runs on each host.
- The robotic control and therefore the robotic control process, `t1dcd`, is on host A.

The NetBackup Device Manager services on host A and B start `t1dd`. The `t1dd` process on host A also starts `t1dcd`. Requests to mount tapes from host B go to `t1dd` on host B, which then sends the robotic command to `t1dcd` on host A.

See “[NetBackup robot types](#)” on page 80.

Oracle StorageTek ACSLS robots

This chapter includes the following topics:

- [About Oracle StorageTek ACSLS robots](#)
- [Sample ACSLS configurations](#)
- [Media requests for an ACS robot](#)
- [About configuring ACS drives](#)
- [Configuring shared ACS drives](#)
- [Adding tapes to ACS robots](#)
- [About removing tapes from ACS robots](#)
- [Robot inventory operations on ACS robots](#)
- [NetBackup robotic control, communication, and logging](#)
- [ACS robotic test utility](#)
- [Changing your ACS robotic configuration](#)
- [ACS configurations supported](#)
- [Oracle StorageTek ACSLS firewall configuration](#)

About Oracle StorageTek ACSLS robots

Note: If you use the access control feature of Oracle StorageTek ACSLS controlled robots and the NetBackup media sharing feature, do the following: ensure that all servers in the NetBackup media server share group have the same ACSLS permissions to all the same ACSLS media and ACSLS drives. Any mismatches can cause failed jobs and stranded tapes in drives.

Oracle StorageTek Automated Cartridge System Library Software controlled robots are NetBackup robot type ACS.

ACS robots are API robots (a NetBackup robot category in which the robot manages its own media).

Unlike other robot types, NetBackup does not track slot locations for the media in ACS robots. The Automated Cartridge System Library Software tracks slot locations and reports them to NetBackup.

The term automated cartridge system (ACS) can refer to any of the following:

- A type of NetBackup robotic control.
- The Oracle StorageTek system for robotic control.
- The highest-level component of the Oracle StorageTek ACSLS. It refers to one robotic library or to multiple libraries that are connected with a media pass-through mechanism.

The ACS library software component can be either of the following Oracle StorageTek products:

- Oracle StorageTek Automated Cartridge System Library Software (ACSLS)
- Oracle StorageTek Library Station

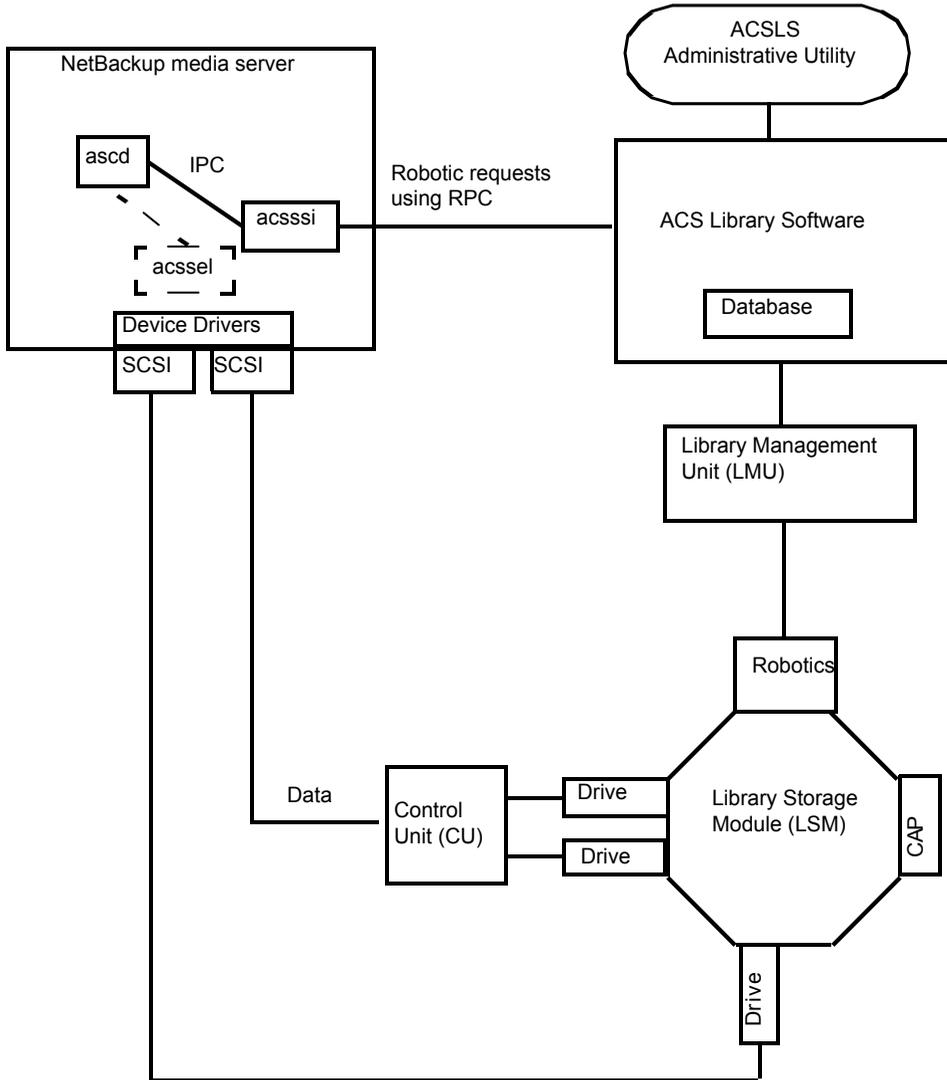
Sample ACSLS configurations

The sample ACSLS configurations show the following:

- A typical UNIX ACSLS configuration.
See [Figure 8-1](#) on page 90.
- A typical Windows ACSLS configuration.
See [Figure 8-2](#) on page 91.
- The major components in typical configurations.
See [Table 8-1](#) on page 92.

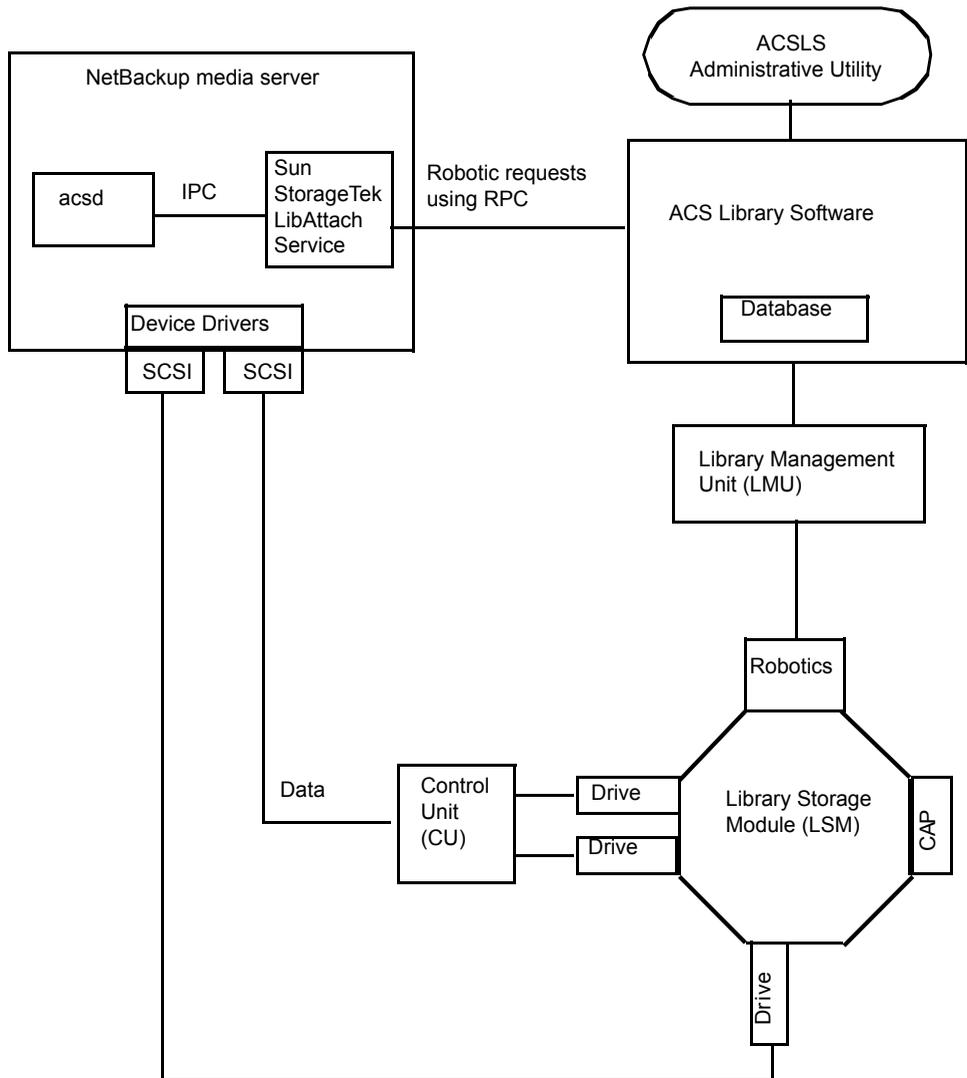
The following figure shows a typical UNIX ACSLS configuration.

Figure 8-1 Typical ACSLS configuration on UNIX



The following figure shows a typical Windows ACSLS configuration.

Figure 8-2 Typical ACSLS configuration on Windows



The following table describes the components of the ACSLS configuration.

Table 8-1 ACSLS configuration component description

Component	Description
NetBackup media server	<p>Specifies a host that has NetBackup media server software and is a client to the ACS library software host.</p> <p>The NetBackup ACS robotic daemon (<code>acsrd</code>) formulates requests for mounts, unmounts, and inventories. An API then uses IPC communication to routes these requests to:</p> <ul style="list-style-type: none"> ■ (UNIX) The NetBackup ACS storage server interface (<code>acsss_i</code>). The requests are converted into RPC-based communications and sent to the ACS library software. ■ (Windows) the Oracle StorageTek LibAttach service. This service sends the requests to the ACS library software.
Oracle StorageTek LibAttach Service Windows computers only	<p>Specifies that Library Attach for Windows, an ACS library software client application, enables Windows servers to use the StorageTek Nearline enterprise storage libraries.</p> <p>LibAttach provides the connection between Windows and ACS library software through a TCP/IP network.</p> <p>Obtain the appropriate LibAttach software from Oracle. See the Veritas support Web site for the latest compatibility information.</p>
The following ACS library software: <ul style="list-style-type: none"> ■ Automated Cartridge System Library Software (ACSL) ■ Sun StorageTek Library Station 	<p>Receives the robotic requests from NetBackup and uses the Library Management Unit to find and mount or unmount the correct cartridge on media management requests.</p> <p>On compatible host platforms, you may be able to configure ACS library software and NetBackup media server software on the same host.</p>
Library Management Unit (LMU)	Provides the interface between the ACS library software and the robot. A single LMU can control multiple ACSLS robots.
Library Storage Module (LSM)	Contains the robot, drives, or media.
Control Unit (CU)	<p>Specifies that the NetBackup media server connects to the drives through device drivers and a control unit (tape controller). The control unit may have an interface to multiple drives. Some control units also allow multiple hosts to share these drives.</p> <p>Most drives do not require a separate control unit. In these cases, the media server connects directly to the drives.</p>
CAP	Specifies the Cartridge Access Port.

Media requests for an ACS robot

The following is the sequence of events for a media request for an ACS robot:

- The Media Manager device daemon (UNIX) or NetBackup Device Manager service (Windows) `ltid` receives the request from `bptm`.
- `ltid` sends a mount request to the NetBackup ACS process `acsd`.
- `acsd` formulates the request.

An API then uses Internal Process Communications (IPC) to send the request on the following systems:

- UNIX. The NetBackup ACS storage server interface `acsssi`. The request is then converted into RPC-based communications and sent to the ACS library software.
- Windows. The Oracle StorageTek LibAttach service. This service sends the request to the ACS library software.
- If the Library Storage Module (LSM) in which the media resides is offline, the ACS library software reports this offline status to NetBackup. NetBackup assigns the request a pending status. NetBackup retries the request hourly until the LSM is online and the ACS library software can satisfy the media request.
- The ACS library software locates the media and sends the necessary information to the Library Management Unit (LMU).
- The LMU directs the robotics to mount the media in the drive. When the LibAttach service (Windows) or `acsssi` (UNIX) receives a successful response from the ACS library software, it returns the status to `acsd`.
- The `acsd` child process (that is associated with the mount request) scans the drive. When the drive is ready, `acsd` sends a message to `ltid` that completes the mount request. NetBackup then begins to send data to or read data from the drive.

About configuring ACS drives

An ACS robot supports DLT or 1/2-inch cartridge tape drives. If an ACS robot contains more than one type of DLT or 1/2-inch cartridge tape drive, you can configure an alternate drive type. Therefore, there can be up to three different DLT and three different 1/2-inch cartridge drive types in the same robot. If you use alternate drive types, configure the volumes by using the same alternate media type. Six drive types are possible: DLT, DLT2, DLT3, HCART, HCART2, and HCART3.

Before you configure drives in NetBackup, configure the operating system tape drivers and device files for those drives. For information about how to do so, refer to the operating system documentation. For guidance about the NetBackup requirements, see the information about the host operating system in this guide.

Use the same methods to create or identify device files for these drives as for other drives. If the drives are SCSI and connect to the robot through a shared control unit, the drives share the same SCSI ID. Therefore, you must specify the same logical unit number (LUN) for each drive.

When you configure ACS drives as robotic in NetBackup, you must include the ACS drive coordinate information.

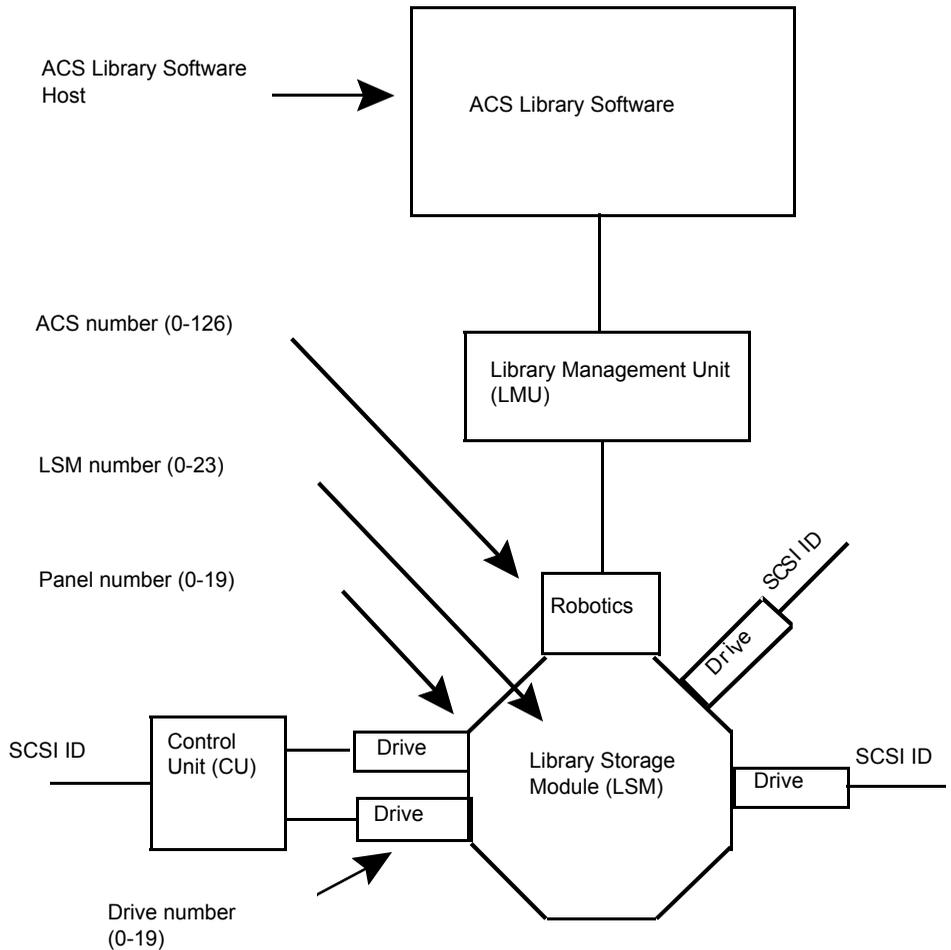
The following table shows the ACS drive coordinates.

Table 8-2 ACS drive coordinates

ACS drive coordinate	Description
ACS number	Specifies the index, in ACS library software terms, that identifies the robot that has this drive.
LSM number	Specifies the Library Storage Module that has this drive.
Panel number	Specifies the panel where the drive is located.
Drive number	Specifies the physical number of the drive in ACS library software terms.

The following figure shows the location of this information in a typical ACS robot.

Figure 8-3 ACSLS robot and drive configuration information



Configuring shared ACS drives

If the ACSLS server does not support serialization, use the following procedure to configure shared drives. Shared drives require the NetBackup Shared Storage Option license. Oracle StorageTek ACSLS versions before 6.1 do not support serialization.) If the server supports serialization, use the NetBackup **Device Configuration Wizard** to configure shared drives.

This procedure can significantly reduce the amount of manual configuration that is required in an SSO environment. For example, for 20 drives that 30 hosts share,

these configuration steps require that you configure only 20 device paths rather than 600 device paths.

During the setup phase, the NetBackup **Device Configuration Wizard** tries to discover the tape drives available. The wizard also tries to discover the positions of the drives within the library (if the robot supports serialization).

A SAN (including switches rather than direct connection) can increase the possibility of errors. If errors occur, you can define the tape drive configuration manually by using the **NetBackup Administration Console** or NetBackup commands.

Take care to avoid any errors. With shared drives, the device paths must be correct for each server. Also, ensure that the drives are defined correctly to avoid errors. (A common error is to define a drive as ACS index number 9 rather than ACS index 0.)

Use the following procedure to configure shared drives in a nonserialized configuration.

To configure shared drives in a nonserialized configuration

- 1 Run the NetBackup **Device Configuration Wizard** on one of the hosts to which drives in an ACS-controlled library are attached. Allow the drives to be added as stand-alone drives.
- 2 Add the ACS robot definition and update each drive to indicate its position in the robot. Make each drive robotic and add the ACS, LSM, Panel, and Drive information.

Information about how to determine the correct drive addresses and how to verify the drive paths is available. See "Correlating device files to physical drives" in the [NetBackup Administrator's Guide, Volume I](#).

- 3 After you verify the drive paths on one host, run the **Device Configuration Wizard** again. Scan all hosts that have ACS drives in the library.

The wizard adds the ACS robot definition and the drives to the other hosts and uses the correct device paths.

For this process to work correctly, the following must be true:

- The wizard discovered the devices and their serial numbers successfully the first time.
- You configured the drive paths correctly on the first host.

Adding tapes to ACS robots

ACS robotic control software supports the following characters in a volume ID that are not valid NetBackup media ID characters. (Volume ID is the ACS term for media ID).

Therefore, do not use any of the following characters when you configure ACS volumes:

- Dollar sign (\$)
- Pound sign (#)
- The yen symbol
- Leading and trailing spaces

The following tables is an overview of how to add tapes to an ACS robot and then add those tapes to NetBackup.

Table 8-3 Adding tapes to ACS robots process

Task	Description
Add barcode labels to the media and insert the media into the robot by using the media access port.	The Library Manager reads the bar codes and classifies the media by media type. A category is assigned to each volume. Some volume categories restrict application access to certain volumes. The Library Manager tracks volume locations.
Define the media in NetBackup by using the ACS volume IDs as media IDs.	<p>To define the media, do one of the following:</p> <ul style="list-style-type: none"> ■ Update the volume configuration by using the robot inventory function. ■ Add new volumes by using the Volume Configuration Wizard. <p>See the <i>NetBackup Administrator's Guide, Volume I</i>: http://www.veritas.com/docs/DOC5332</p> <p>Because the ACS volume IDs and bar codes are the same, NetBackup has a record of the bar codes for the media. Note that you do not enter slot locations because the ACS library software manages slot locations.</p>
Verify the volume configuration	Use Show Contents and Compare Contents with Volume Configuration from the Robot Inventory dialog.

About removing tapes from ACS robots

You can remove tapes by using the Sun StorageTek utility or by using NetBackup.

See ["Removing tapes using the ACSLS utility"](#) on page 98.

See “[Removing tapes using NetBackup](#)” on page 98.

Removing tapes using the ACSLS utility

If you remove media from an ACS robot, you must move the media logically to stand alone in NetBackup.

If you do not move media logically, NetBackup does not know that the media were moved. NetBackup may issue mount requests for it, which causes a misplaced tape error.

However, you can move media from one location to another within the robot. The ACS library software finds the requested media if its database is current.

To remove tapes using the SCSLS utility

- ◆ Do one of the following:
 - Update the volume configuration by using the NetBackup robot inventory function.
See the *NetBackup Administrator's Guide, Volume I*.
<http://www.veritas.com/docs/DOC5332>
 - Move the volumes.
See the *NetBackup Administrator's Guide, Volume I*.
<http://www.veritas.com/docs/DOC5332>

Removing tapes using NetBackup

To remove tapes using NetBackup

- ◆ Use one of the following methods:
 - Select **Actions > Eject Volumes From Robot** in the **NetBackup Administration Console**.
 - Use the NetBackup `vmchange` command.
See the *NetBackup Commands Reference Guide*.
<http://www.veritas.com/docs/DOC5332>

Both of these methods performs the logical move and the physical move.

Robot inventory operations on ACS robots

If the ACS library software host is a Sun StorageTek Library Station, an Inventory Robot Filter (`INVENTORY_FILTER`) entry may be required in the `vm.conf` file. Old versions of Library Station do not support queries of all volumes in an ACS robot.

In NetBackup, the ACS robot type supports bar codes.

The following sequence of events occurs when you inventory an ACS robot in NetBackup:

- NetBackup requests volume information from the ACS library software.
- The ACS library software provides a listing of the volume IDs, media types, ACS location, and LSM location from its database.
 See [Table 8-4](#) on page 99.
- NetBackup maps the volume IDs into media IDs and bar codes. For example in the previous table, volume ID 100011 becomes media ID 100011 and the barcode for that media ID is also 100011.
- If the operation does not require a volume configuration update, NetBackup uses the media type defaults for ACS robots when it creates its report.
- If the operation requires a volume configuration update, NetBackup does the following:
 - Maps the ACS media types to the default NetBackup media types.
 - Adds the ACS and the LSM locations for new volumes to the EMM database. This location information is used for media and drive selection.

Information about the default media type mappings and how to configure media type mappings is available.

See the [NetBackup Administrator's Guide, Volume I](#).

The following table shows an example of the ACS drive coordinates that NetBackup receives.

Table 8-4 ACS drive coordinates

ACS volume ID	ACS media type	ACS	LSM
100011	DLTIV	0	0
200201	DD3A	0	0
412840	STK1R	0	1
412999	STK1U	0	1
521212	JLABEL	0	0
521433	STK2P	0	1
521455	STK2W	0	1
770000	LTO_100G	0	0

Table 8-4 ACS drive coordinates (*continued*)

ACS volume ID	ACS media type	ACS	LSM
775500	SDLT	0	0
900100	EECART	0	0
900200	UNKNOWN	0	0

Configuring a robot inventory filtering on ACS robots

If you want NetBackup to use only a subset of the volumes under ACS library control, you can filter the volume information from the library. To do so, you use the ACSLS administrative interface to assign the volumes you want to use to a scratch pool or pools. Then you configure NetBackup to use only the volumes in those scratch pools.

A NetBackup robot inventory includes the volumes that exist in the ACS scratch pool. The ACS library software moves each volume from the scratch pool after it is mounted.

A partial inventory also includes those volumes that NetBackup can validate exist in the robotic library, including volumes not in the ACS scratch pool. To prevent losing track of previously mounted volumes, the library reports the complete list of volumes that exist in the robotic library.

The following procedure is an example of how to configure an inventory filter.

To configure an inventory filter (example)

- 1 Use the ACSLS administrative interface (ACSSA) command to create a scratch pool. Assign ID 4 and 0 to 500 as the range for the number of volumes, as follows:

```
ACSSA> define pool 0 500 4
```

- 2 Use the ACSLS administrative interface (ACSSA) command to define the volumes in scratch pool 4:

```
ACSSA> set scratch 4 600000-999999
```

- 3 On the NetBackup media server from which you invoke the inventory operation, add an INVENTORY_FILTER entry to the vm.conf file. The following is the usage statement:

```
INVENTORY_FILTER = ACS robot_number BY_ACS_POOL acs_scratch_pool1  
[acs_scratch_pool2 ...]
```

The following define the options and arguments:

- *robot_number* is the number of the robot in NetBackup.
- *acs_scratch_pool1* is the scratch pool ID as configured in the ACS library software.
- *acs_scratch_pool2* is a second scratch pool ID (up to 10 scratch pools are allowed).

For example, the following entry forces ACS robot number 0 to query scratch volumes from Sun StorageTek pool IDs 4 and 5.

```
INVENTORY_FILTER = ACS 0 BY_ACS_POOL 4 5
```

NetBackup robotic control, communication, and logging

How NetBackup uses robotic control, communication, and logging during tape operations depends on the operating system type as follows:

- Windows systems
See [“NetBackup robotic control, communication, and logging for Windows systems”](#) on page 101.
- UNIX systems
See [“NetBackup robotic control, communication, and logging for UNIX systems”](#) on page 102.

NetBackup robotic control, communication, and logging for Windows systems

The NetBackup `acsd` process provides robotic control to mount and dismount volumes. It also requests inventories of the volumes that are under the control of ACS library software. The NetBackup Device Manager service `ltid` starts the `acsd` process and communicates with it.

The `acsd` process requests SCSI tape unloads through the device host's tape driver before it uses the ACS API to request that tape dismounts. This request process accommodates the configurations that have SCSI multiplexors. Loaded tapes are not ejected forcibly when a dismount operation occurs.

NetBackup robotic control, communication, and logging for UNIX systems

On UNIX systems, several NetBackup daemons and processes provide robotic control, communication, and logging.

NetBackup ACS daemon (acsd)

The NetBackup ACS daemon `acsd` provides robotic control to mount and dismount volumes. It also requests inventories of the volumes that are under the control of ACS library software. The Media Manager device daemon `ltid` starts the `acsd` daemon and communicates with it. If `ltid` is active already, you can start `acsd` manually.

The `acsd` daemon requests SCSI tape unloads through the device host's tape driver before it uses the ACS API to request that tape dismounts. This control process accommodates the configurations that have SCSI multiplexors. Loaded tapes are not ejected forcibly when a dismount operation occurs.

When `acsd` starts, it first starts the NetBackup `acssel` process and then starts the `acsssi` process. When it starts `acsssi`, `acsd` passes the ACS library software host name to `acsssi`. One copy of `acsssi` starts for each ACS library software host that appears in the NetBackup device configuration for the media server. If multiple media servers share drives in an ACS robot, `acsssi` must be active on each media server.

NetBackup ACS SSI event logger (acssel)

The NetBackup ACS storage server interface (SSI) event logger `acssel` is modeled after the Sun StorageTek `mini_el` event logger. Therefore, its functional model differs from other NetBackup robotic controls.

The NetBackup `acsd` daemon starts `acssel` automatically. You also can start it manually. Event messages are logged to the following file:

```
/usr/opensv/volmgr/debug/acsssi/event.log
```

Note: Veritas recommends that `acssel` run continuously because it tries to connect on the event logger's socket for its message logging. If `acsssi` cannot connect to `acssel`, NetBackup cannot process requests immediately. Therefore, retry and error recovery situations can occur.

On UNIX systems, only the `kill` command stops `acsssel`. The NetBackup `bp.kill_all` utility (UNIX) stops the `acsssel` process. On Windows systems, the `bpdown.exe` program stops the `acsssel` process.

The full path to the event logger is `/usr/opensv/volmgr/bin/acsssel`. The usage format is as follows:

```
acsssel [-d] -s socket_name
```

The following describes the options:

- `-d` displays debug messages (by default, debug messages are disabled).
- `socket_name` is the socket name (or IP port) to listen on for messages.

Using `acsssel` with a different socket name

If the `vm.conf` file does not contain an `ACS_SEL_SOCKET` entry, `acsssel` listens on socket name 13740 by default.

You can change this default by using one of the following methods:

- Modify the `vm.conf` configuration file.
See [To change the default by modifying the `vm.conf` configuration file](#).
- Add environment variables. This method assumes that one ACS robot is configured and that the SSI default socket name has not been changed. (The `vm.conf ACS_SEL_SOCKET` entry can change the default).
See [To change the default by adding environment variables](#).
`acsssel` also has a command line option to specify the socket name. However, because `acsssi` needs to know the event logger socket name, setting an environment variable is preferred.

To change the default by modifying the `vm.conf` configuration file

- 1 Edit the `vm.conf` file and add an `ACS_SEL_SOCKET` entry. The following is an example:

```
ACS_SEL_SOCKET = 13799
```

- 2 Stop the `acsdd`, `acsssi`, and `acsssel` processes by invoking the following script. (This script stops all NetBackup processes.)

```
/usr/opensv/NetBackup/bin/bp.kill_all
```

- 3 Restart the NetBackup daemons and processes by invoking the following script:

```
/usr/opensv/NetBackup/bin/bp.start_all
```

To change the default by adding environment variables

- 1 Stop the `acsd`, `acsssi`, and `acsssel` processes by invoking the following script. (This script stops all NetBackup processes.)

```
/usr/opensv/NetBackup/bin/bp.kill_all
```

- 2 Set the wanted socket name in an environment variable and export it. The following is an example:

```
ACS_SEL_SOCKET = 13799  
export ACS_SEL_SOCKET
```

- 3 Start the event logger in the background.

```
/usr/opensv/volmgr/bin/acsssel &
```

- 4 Set the ACS library software host name for `acsssi` in an environment variable.

```
CSI_HOSTNAME = einstein  
export CSI_HOSTNAME
```

- 5 Start `acsssi` as follows:

```
/usr/opensv/volmgr/bin/acsssi 13741 &
```

- 6 Optionally, start `acstest` by using the `robtest` utility or by using the following command:

```
/usr/opensv/volmgr/bin/acstest -r einstein -s 13741
```

If you request SCSI unloads, you also must specify drive paths on the `acstest` command line.

See [“ACS robotic test utility”](#) on page 106.

The `robtest` utility specifies drive paths automatically if ACS drives have been configured.

- 7 Start `ltid` as follows, which starts `acsd`. You can use the `-v` option for verbose message output.

```
/usr/opensv/volmgr/bin/ltid
```

During initialization, `acsd` obtains the SSI Event Logger socket name from `vm.conf` and sets `ACS_SEL_SOCKET` in the environment before it starts `acsssel`. If `acsssi` is started manually, it has to use (listen on) the same SSI socket that `acsd` uses to send data.

NetBackup ACS storage server interface (acsssi)

The NetBackup ACS storage server interface (SSI) `acsssi` communicates with the ACS library software host. `acsssi` processes all RPC communications from `acsd` or from the ACS robotic test utility that are intended for the ACS library software.

One copy of `acsssi` must run for each unique ACS library software host that is configured on a NetBackup media server. `acsd` tries to start copies of `acsssi` for each host. However, if an `acsssi` process for a specific ACS library software host exists already, the new `acsssi` processes for that host fails during initialization.

In normal operations, `acsssi` runs in the background and sends log messages to `acsse1`.

You can specify the socket name (IP port) used by `acsssi` in any of the following ways:

- On the command line when you start `acsssi`.
- By using an environment variable (`ACS_SSI_SOCKET`).
- Through the default value.

If you configure `acsssi` to use a nondefault socket name, you also must configure the ACS daemon and ACS test utility to use the same socket name.

The ACS library software host name is passed to `acsssi` by using the `CSI_HOSTNAME` environment variable.

`acsssi` is based on the Sun StorageTek storage server interface. Therefore, it supports environment variables to control most aspects of operational behavior.

See [“Optional environment variables”](#) on page 106.

About the ACS_SSI_SOCKET configuration option

By default, `acsssi` listens on unique, consecutive socket names; the socket names begin at 13741. To specify socket names on an ACS library software host basis, you can add a configuration entry in the NetBackup `vm.conf` file.

Use the following format:

```
ACS_SSI_SOCKET = ACS_library_software_hostname socket_name
```

The following is an example entry (do not use the IP address of the ACS library host for this parameter):

```
ACS_SSI_SOCKET = einstein 13750
```

Starting acsssi manually

This method is not the recommended method to start `acsssi`. Normally, `acsd` starts `acsssi`.

Before you can start `acsssi` manually, you must configure the `CSI_HOSTNAME` environment variable. The following is a Bourne shell example:

```
CSI_HOSTNAME=einstein
export CSI_HOSTNAME
/usr/opencv/volmgr/bin/acsssi 13741 &
```

Use the following procedure to start `acsssi`.

To start `acsssi`

- 1 Start the event logger, `acsse1`.
- 2 Start `acsssi`. The format is `acsssi socket_name`.

Optional environment variables

If you want individual NetBackup `acsssi` processes to operate differently, you can set environment variables before the `acsssi` processes are started.

The following table describes the optional environment variables.

Table 8-5 Optional environment variables

Environment variable	Description
<code>SSI_HOSTNAME</code>	Specifies the name of the host where ACS library software RPC return packets are routed for ACS network communications. By default, the local host name is used.
<code>CSI_RETRY_TIMEOUT</code>	Set this variable to a small positive integer. The default is 2 seconds.
<code>CSI_RETRY_TRIES</code>	Set this variable to a small positive integer. The default is five retries.
<code>CSI_CONNECT_AGETIME</code>	Set this variable to a value between 600 seconds and 31536000 seconds. The default is 172800 seconds.

ACS robotic test utility

The `acstest` utility lets you verify ACS communications and provides a remote system administrative interface to an ACS robot. It can also be used to query, enter,

eject, mount, unload, and dismount volumes. In addition, `acstest` lets you define, delete, and populate ACS library software scratch pools.

While `acsd` services requests, do not use `acstest`. Communication problems may occur if `acsd` and `acstest` process ACS requests at the same time.

acstest on Windows systems

`acstest` depends on the Sun StorageTek LibAttach service being started successfully. You can verify that this service is started by using the Services tool available in administrative tools in the Windows control panel. `acstest` attempts to communicate with ACS library software by using the `LibAttach` service.

The usage format follows:

```
acstest -r ACS_library_software_hostname [-d device_name ACS, LSM, panel, drive] ... [-C sub_cmd]
```

The following example assumes that the `LibAttach` service started:

```
install_path\Volmgr\bin\acstest -r einstein -d Tape0 0,0,2,1
```

acstest on UNIX systems

`acstest` depends on `acsssi` being started successfully. You can use the UNIX `netstat -a` command to verify that a process listens on the SSI socket. `acstest` attempts to communicate with ACS library software using `acsssi` and connects on an existing socket.

The usage format follows. You can pass the socket name on the command line. Otherwise, the default socket name (13741) is used.

```
acstest -r ACS_library_software_hostname [-s socket_name] [-d drive_path ACS, LSM, panel, drive] ... [-C sub_cmd]
```

The following example assumes that the `acsssi` process has been started by using socket 13741:

```
/usr/opencv/volmgr/bin/acstest -r einstein -s 13741
```

Changing your ACS robotic configuration

UNIX and Linux systems only.

If you change your ACS robot configuration, you should update NetBackup so that `acsssi` can successfully communicate with `acsd`, `acstest`, and ACS library software.

Any `acsssi` processes must be canceled after your changes are made and before the Media Manager device daemon `ltid` is restarted. Also, for the `acstest` utility to function, `acsssi` for the selected robot must be running.

Use the following procedure to update NetBackup after you change your configuration.

To update NetBackup after you change your configuration

- 1 Make your configuration changes.
- 2 Use `/usr/opensv/NetBackup/bin/bp.kill_all` to stop all running processes.
- 3 Restart the NetBackup daemons and processes by invoking the following script:

```
/usr/opensv/NetBackup/bin/bp.start_all
```

ACS configurations supported

UNIX and Linux systems only.

NetBackup supports the following ACS configurations:

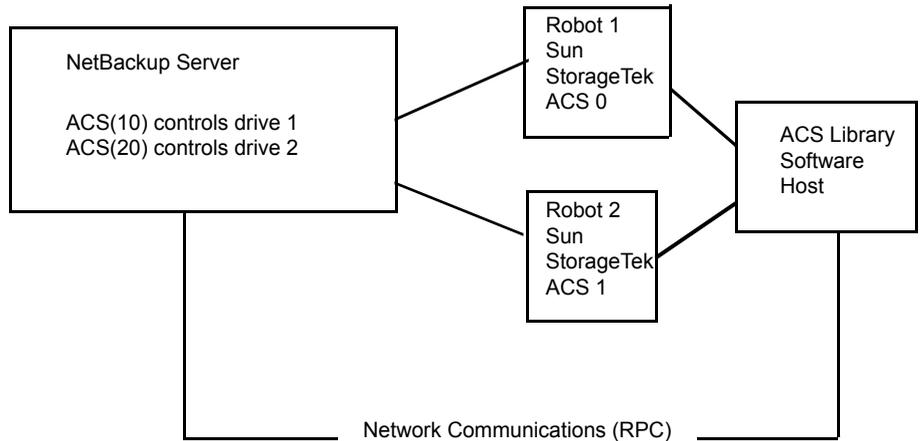
- Multiple robots that are controlled from a single ACS host
See [“Multiple ACS robots with one ACS library software host”](#) on page 108.
- Multiple robots that are controlled from multiple ACS hosts
See [“Multiple ACS robots and ACS library software hosts”](#) on page 109.

Multiple ACS robots with one ACS library software host

NetBackup supports the following configuration:

- A NetBackup server is connected to drives in multiple ACS robots.
- The robots are controlled from a single ACS library software host.

The following figure shows multiple ACS robots that are controlled from a single ACS library software host.

Figure 8-4 Multiple ACS robots, one ACS library software host

Inventory requests include: the volumes that are configured on the ACS library software host that resides on the ACS robot that is designated in the drive address.

In this example, assume the following about drive 1:

- Has an ACS drive address (ACS, LSM, panel, drive) of 0,0,1,1 in the NetBackup device configuration
- Is under control of robot number 10 (ACS(10)).

If any other robot ACS(10) drives have a different ACS drive address (for example, 1,0,1,0), the configuration is invalid.

NetBackup supports configurations of multiple LSMs in a single ACS robot if a pass-through port exists.

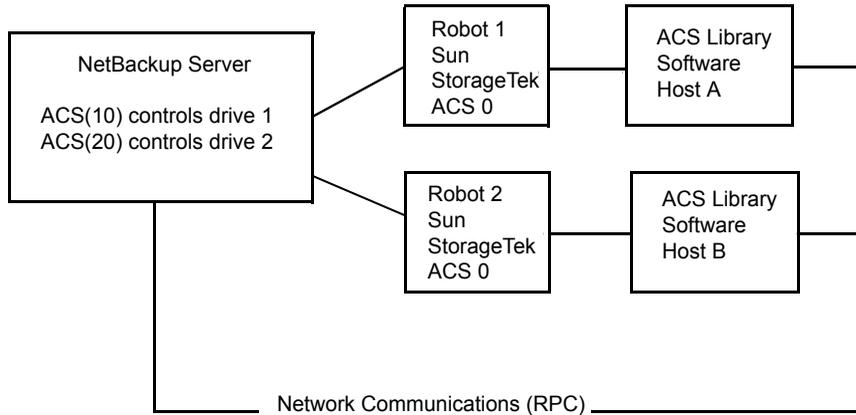
Multiple ACS robots and ACS library software hosts

NetBackup supports the following configuration:

- A NetBackup server is connected to drives in multiple ACS robots.
- The robots are controlled from separate ACS library software hosts.

The following figure shows multiple ACS robots that are controlled from multiple ACS library software hosts.

Figure 8-5 Multiple ACS robots, multiple ACS library software hosts



Inventory requests include the volumes that are configured on the ACS library software hosts (Host A for Robot 1 and Host B for Robot 2). The software hosts reside on the robot (ACS 0 for each) that is designated in the Sun StorageTek drive address.

In this example, assume the following about drive 1:

- Has an ACS drive address (ACS, LSM, panel, drive) of 0,0,1,1 in the NetBackup device configuration
- Is under control of robot number 10 (ACS(10))

If any other robot ACS(10) drives have a different ACS drive address (for example, 1,0,1,0), the configuration is invalid.

NetBackup supports configurations of multiple LSMs in a single ACS robot if a pass-through port exists.

Oracle StorageTek ACSLS firewall configuration

To configure an ACS robot in an Sun StorageTek ACSLS firewall environment, use the following NetBackup `vm.conf` file configuration entries to designate TCP port connections:

- `ACS_CSI_HOSTPORT`
- `ACS_SSI_INET_PORT`
- `ACS_TCP_RPCSERVICE`

More information about `vm.conf` entries is available.

See the [NetBackup Administrator's Guide, Volume I](#).

The Sun StorageTek ACSLS server configuration options must match the entries in the `vm.conf` file. For example, in a typical ACSLS firewall configuration, you would change the following settings as shown:

- `Changes to alter use of TCP protocol...`
Set to `TRUE` - Firewall-secure ACSLS runs across TCP.
- `Changes to alter use of UDP protocol...`
Set to `FALSE` - Firewall-secure ACSLS runs across TCP.
- `Changes to alter use of the portmapper...`
Set to `NEVER` - Ensures that the ACSLS server does not query the portmapper on the client platform.
- `Enable CSI to be used behind a firewall...`
Set to `TRUE` - Allows specification of a single port for the ACSLS server.
- `Port number used by the CSI...`
The port that the user chooses. The 30031 default value is used most often. This port number must match the port number that you specify in the NetBackup `vm.conf` file.

For complete information about setting up a firewall-secure ACSLS server, refer to your vendor documentation.

Device configuration examples

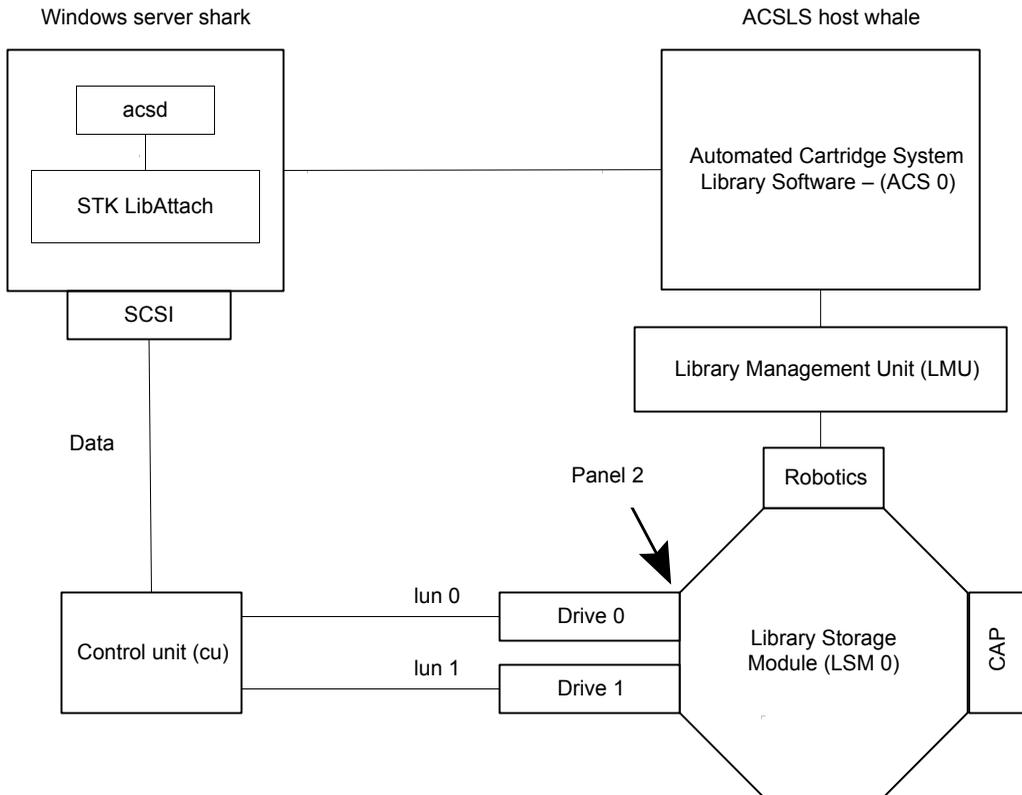
This chapter includes the following topics:

- [An ACS robot on a Windows server example](#)
- [An ACS robot on a UNIX server example](#)

An ACS robot on a Windows server example

The following figure shows a Windows server and ACS robot configuration.

Figure 9-1 Windows server and ACS robot configuration example



This configuration uses an Automated Cartridge System (ACS) robot for storage. Server shark can be a Windows NetBackup master server or media server.

The following are items to note when you review this example:

- The Oracle StorageTek ACSLS host (in the Add Robot dialog) is host whale, where the ACS library software resides. In this example, Automated Cartridge System Library Software (ACSL) is installed as the ACS library software. On some server platforms, you can run NetBackup media server software and ACS library software on the same server. Therefore, you need only one server.
- The ACS, LSM, PANEL, and DRIVE numbers are part of the ACS library software configuration and must be obtained from the administrator of that host.
- Robot number and ACS number are different terms. Robot number is the robot identifier used in NetBackup. ACS number is the robot identifier in ACS library software. These numbers can be different, although they both default to zero.

- If you connect the drives through an independent control unit, you must use the correct Logical Unit Numbers (LUNs) so that the correct tape name is used.
- The Add Robot dialog entries include an ACSLS Host entry so that the ACS library software host communicates by using STK LibAttach software. This software must be installed on each Windows server that has the ACS drives attached to it.

The following table shows the robot attributes for the remote host shark.

Table 9-1 Add Robot dialog entries (remote host)

Dialog box field	Value
Device Host	shark
Robot Type	ACS (Automated Cartridge System)
Robot Number	0
Robot control is handled by a remote host	Set (cannot be changed for this robot type)
ACSLS Host	whale

The following table shows the drive 0 attributes.

Table 9-2 Add Drive dialog entries (drive 0)

Dialog box field	Value
Device Host	shark
Drive Type	1/2" Cartridge (hcart)
Drive Name	shark_drive_0
Path Information	[5,0,1,0]
Drive is in a Robotic Library	Yes
Robotic Library	ACS(0) - whale
ACS	ACS: 0 LSM: 0 PANEL: 2 DRIVE: 0

The following table shows the drive attributes for drive 1.

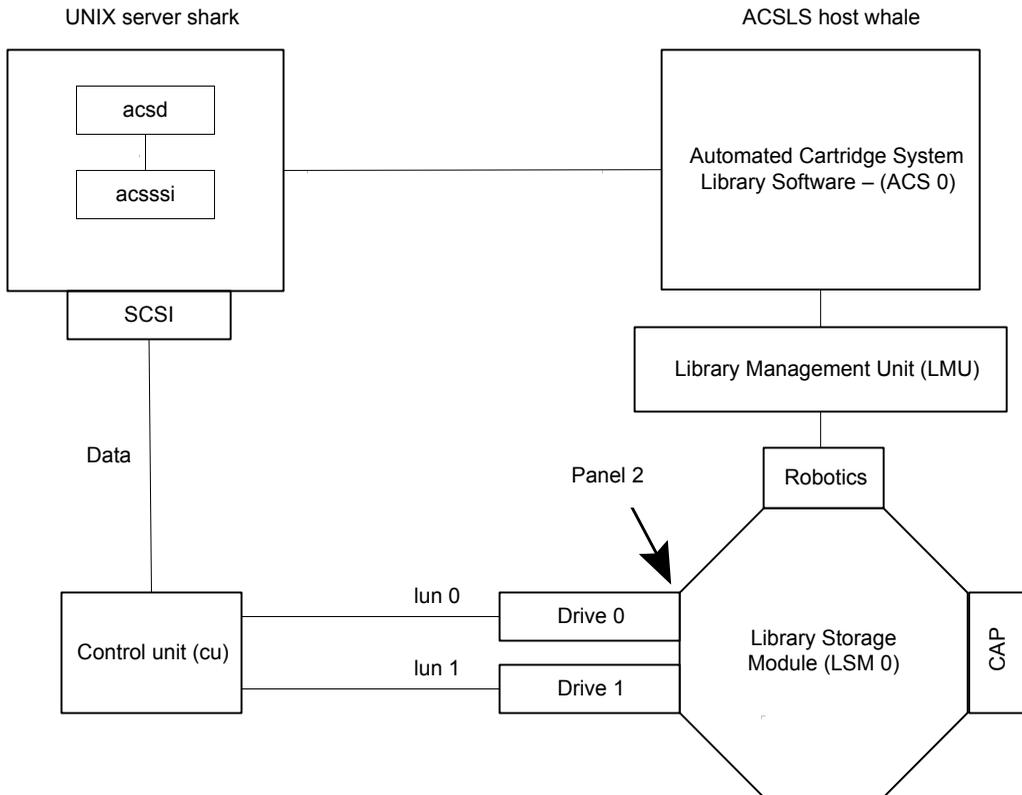
Table 9-3 Add Drive dialog entries (drive 1)

Dialog box field	Value
Device Host	shark
Drive Type	1/2" Cartridge (hcart)
Drive Name	shark_drive_1
Path Information	[4,0,1,1]
Drive is in a Robotic Library	Yes
Robotic Library	ACS(0) - whale
ACS	ACS: 0 LSM: 0 PANEL: 2 DRIVE: 1

An ACS robot on a UNIX server example

The following figure shows a UNIX server and ACS robot configuration.

Figure 9-2 UNIX server and ACS robot configuration example



This configuration uses an Automated Cartridge System (ACS) robot for storage. Host shark can be a UNIX NetBackup master server or media server.

The following are some items to note when you review this example:

- The ACSLS Host (in the Add Robot dialog) is server whale, where the ACS library software resides. In this example, Automated Cartridge System Library Software (ACSL) is installed as the ACS library software. On some server platforms, you can run NetBackup media server software and ACS library software on the same server. Therefore, you need only one server.
- The ACS, PANEL, LSM, and DRIVE numbers are part of the ACS library software configuration and must be obtained from that system.
- Robot number and ACS number are different terms. Robot number is the robot identifier used in NetBackup. ACS number is the robot identifier in ACS library software. These numbers can be different, although they both default to zero.

- If you connect the drives through an independent control unit, you must use the correct Logical Unit Numbers (LUNs) so that the correct tape name is used.
- The Add Robot dialog entries include an ACSLS Host entry. That entry configures NetBackup to use the ACS Storage Server Interface (acsssi) to communicate with the ACS library software host.

The following table shows the robot attributes.

Table 9-4 Add Robot dialog entries (remote host)

Dialog box field	Value
Device Host	shark
Robot Type	ACS (Automated Cartridge System)
Robot Number	0
Robot control is handled by a remote host	Set (cannot be changed for this robot type)
ACSLS Host	whale

The following table shows the drive 0 attributes.

Table 9-5 Add Drive dialog entries (drive 0)

Dialog box field	Value
Device Host	shark
Drive Name	shark_drive_0
Drive Type	1/2" Cartridge (hcart)
No Rewind Device	/dev/rmt1.1
Drive is in a Robotic Library	Yes
Robotic Library	ACS(0) - whale
ACS	ACS Number: 0 LSM Number: 2 PANEL Number: 0 DRIVE Number: 0

The following table shows the drive 1 attributes.

Table 9-6 Add Drive dialog entries (drive 1)

Dialog box field	Value
Device Host	shark
Drive Name	shark_drive_1
Drive Type	1/2" Cartridge (hcart)
No Rewind Device	/dev/rmt1.1
Drive is in a Robotic Library	Yes
Robotic Library	ACS(0) - whale
ACS	ACS Number: 0 LSM Number: 2 PANEL Number: 0 DRIVE Number: 1

Index

A

- ACS. *See* Automated Cartridge System
- ACS daemon (acsd)
 - NetBackup 102
- ACS drives
 - configuring 93
- ACS robot
 - ACSLs firewall configuration 110
 - changing configuration 107
- ACS robot on a UNIX server
 - configuration example 116
- ACS robot on a Windows server
 - configuration example 113
- ACS robot type 81
- ACS robotic test utility 107
- ACS robots 81
 - adding tapes 97
 - removing tapes 97
 - robot inventory filtering 100
 - robot inventory operations 99
 - with multiple ACS hosts 109
 - with single ACS host 108
- ACS SSI event logger (acssel)
 - NetBackup 102
 - using with a different socket name 103
- ACS storage server interface (acsssi)
 - NetBackup 105
 - starting manually 106
- ACS_SSI_SOCKET
 - configuration option 105
- acsd daemon 102
- acsd process
 - NetBackup 101
- ACSLs
 - configurations 89
- ACSLs utility
 - removing tapes 98
- acssel 102
 - using with a different socket name 103
- acsssi 105
 - environment variables 106
- acsssi (*continued*)
 - starting manually 106
- acstest 104, 107
 - on UNIX systems 107
 - on Windows systems 107
- adding tapes
 - to ACS robots 97
- agile addressing 26
- AIX
 - adapter number conventions 13
 - adapter numbers 13
 - command summary 23
 - configuring robotic control device files for IBM robots 14
 - configuring tape drive device files 15
 - disabling SPC-2 SCSI reserve 22
 - introduction 12
 - locate-block 17
 - smit tool 12
 - tape drive configuration
 - extended file marks 16
 - make device files 17
 - multiple densities 21
 - variable mode devices 16
- AL-PA destination ID
 - Solaris 60
- alternate media types
 - ACS robots 93
- API robots 89
- atdd driver
 - HP-UX 45
- attaching devices
 - to a Windows system 78
- attributes
 - robot 81
- Automated Cartridge System
 - adding volumes 97
 - barcode operations 99
 - configuration example 113, 116
 - configurations supported 108
 - Library Server (ACSLs) 89, 92
 - media requests 93

- Automated Cartridge System *(continued)*
 - multiple ACS robots with multiple ACS hosts 109
 - multiple ACS robots with single ACS host 108
 - robot inventory filtering 100
 - Solaris 57
 - special characters 97
 - STK Library Station 89, 92
- Automated Cartridge System (ACS)
 - removing tapes 97

B

- Berkeley-style close
 - HP-UX 27
 - Solaris 71
- boot -r
 - Solaris 76

C

- cfgmgr command 24
- chdev command 16, 24
- choosing
 - tape driver 15
- command summary
 - for AIX 23
 - HP-UX 48
 - Linux 55
 - Solaris 75
- configuration
 - changing ACS robot 107
 - for Sun StorEdge Network Foundation HBA driver 59
- configuration example
 - ACS robot on a UNIX server 116
 - ACS robot on a Windows server 113
- configuration guidelines
 - HP-UX 25
- configuration option
 - ACS_SSI_SOCKET 105
- configurations supported
 - Automated Cartridge System 108
- configuring
 - ACS drives 93
 - legacy device files 35
 - persistent DSFs 30
 - robotic control device files for IBM robots in AIX 14
 - SAN clients to recognize FT media servers 73
 - shared ACS drives 95

- configuring *(continued)*
 - tape drive device files in AIX 15
- control unit
 - ACS 92
- conventions
 - RS/6000 AIX adapter number 13
- creating
 - HP-UX persistent DSFs 30
 - legacy SCSI and FCP robotic controls on HP-UX 36
 - legacy tape drive device files 43
 - no rewind device files for tape drives 17
 - persistent DSF pass-through paths 32
 - sctl device file for FCP (Itanium) 41
 - sctl device file for FCP (PA-RISC) 39
 - sctl device file for SCSI (PA-RISC) 37
 - tape drive pass-through device files 43
- creating device files
 - for SAN clients on AIX 14
 - for SAN clients on HP-UX 35

D

- debug mode
 - for st tape driver 51
- device
 - configuration wizard 95
- device addressing schemes
 - HP-UX 26
- device configuration sequence 9
- device discovery 9
- device drivers
 - for legacy device files 33
 - for persistent DSFs 29
 - sg
 - Linux 50
 - Solaris 57
 - st
 - Linux 51
- device drivers and files
 - for persistent DSFs 28
- device files
 - creating for SAN clients on AIX 14
 - creating for SAN clients on HP-UX 35
 - creating no rewind 17
 - for legacy tape drives 33
 - Linux (2.6 kernel) robotic control 52
 - Linux (2.6 kernel) tape drive 52
 - no rewind 17

- device special files
 - persistent 28
- disable
 - HP-UX EMS Tape Device Monitor for a SAN 47
 - SPC-2 SCSI reserve in HP-UX 47
- disabling
 - Solaris Multiplexed I/O (MPxIO) 60
 - SPC-2 SCSI reserve in AIX 22
 - SPC-2 SCSI reserve on Solaris 72
- driver unloading
 - Solaris 67
- drives
 - Sony S-AIT 22
- drstat command 84
- DSFs. See device special files 28

E

- environment variables
 - for acsssi processes 106
- example
 - robotic process 86
 - sg.conf file 63
 - sg.links file 64
 - st.conf file 63
- example configuration
 - ACSLs 89
- examples
 - no rewind device file 19
 - SCSI and FCP robotic control device files 69
- extended file marks
 - for drives 16

F

- fabric assigned destination ID
 - Solaris 60
- fast-tape positioning. See locate-block
- Fibre Channel
 - drivers 67
 - HP-UX configuration example 39, 41
- fibre channel
 - binding process
 - Solaris 59
- Fibre Channel HBA drivers
 - binding 59
- firewall configuration
 - Sun StorageTek ACSLS 110
- fixed length block 16

- forward-space-file/record
 - HP-UX 28

H

- Hardware Compatibility List (HCL) 80
- HP-UX
 - command summary 48
 - configuration guidelines 25
 - creating legacy SCSI and FCP robotic controls 36
 - creating persistent DSFs 30
 - device addressing schemes 26
 - disable EMS Tape Device Monitor for a SAN 47
 - disable SPC-2 SCSI reserve 47
 - legacy device drivers and files 32
 - robotic control 26
 - SAM utility 47
 - SCSI robotic controls 33
 - SPC-2 SCSI reserve 47
 - Tape drive configuration
 - Berkeley-style close 27
 - make device files 28
 - upgrading NetBackup to use persistent DSFs 31

I

- introduction
 - Linux 49
 - Solaris 56
- ioscan command
 - HP-UX 48

L

- legacy device drivers and files
 - HP-UX 32
- legacy device files
 - configuring 35
 - device drivers supported 33
- legacy pass-through paths
 - for tape drives 34
- legacy tape drive device files
 - creating 43
- legacy tape drives
 - device file names 33
- Library Management Unit 92
- Library Management Unit (LMU) 93
- Library Storage Module 92
- Library Storage Module (LSM) 93
- Linux
 - command summary 55

Linux (*continued*)
 introduction 49
 loading drivers 51
 robotic controls 51
 SAN clients 53
 SCSI robotic controls 51
 sg driver 50
 utilities to test SCSI devices 55
 verifying the device configuration 52

Linux (2.6 kernel)
 robotic controls 52
 tape drive device files 52

LMU. *See* Library Management Unit

locate-block
 on AIX 17
 on Solaris 71

lsattr command 24

lsdev command
 HP-UX 48
 on AIX 23

LSM. *See* Library Storage Module

lsmod command
 Linux 51

M

media requests
 for an ACS robot 93

mknod command
 HP-UX 48

modinfo command
 Solaris 76

modprobe command
 Linux 51

mt command
 Linux 55

multiple tape densities
 using 21

N

NetBackup
 ACS daemon (acsd) 102
 ACS SSI event logger (acssel) 102
 ACS storage server interface (acsssi) 105
 acsd process 101
 removing tapes 98
 robotic control, communication, and logging 101
 using acssel with different socket name 103

NetBackup sg driver
 verify installation 58

no rewind device file example 19

no rewind device files 17
 creating 17
 Solaris 71

nonstandard tape drives 73

O

odmget command 24

Oracle
 configuring shared ACS drives 95
 StorageTek ACSLS robots 89

P

pass-through paths
 for persistent DSFs 29

persistent DSF pass-through paths
 creating 32

persistent DSFs
 configuring 30
 device drivers 29
 device drivers and files 28
 for robotic control 29
 for tape drive access 29
 pass-through paths 29

processes
 by robotic type 85
 robotic 84
 robotic control 84

R

rem_drv command
 Solaris 76

removing
 tapes from ACS robots 97
 tapes using NetBackup 98
 tapes using the ACSLS utility 98

requirements
 tape drive device files 27

robot
 attributes 81
 process 84

robot inventory operations
 on ACS robots 99

robot types 80
 robotic processes 85

- robotic
 - control process 84
 - test utilities 84
 - test utilities, ACS 107
- robotic control
 - for persistent DSFs 29
 - on HP-UX 26
 - on UNIX systems 102
 - on Windows systems 101
- robotic control device files
 - for IBM robots in AIX 14
- robotic control, communication, and logging
 - during tape operations 101
- robotic controls
 - SCSI
 - HP-UX 33
 - Linux (2.6 kernel) 52
 - on AIX 51, 69
 - Solaris 68
- robotic inventory
 - filtering 100
- robotic process
 - example 86
- robotics
 - table-driven 83
- robots
 - ACS 81
 - Oracle StorageTek ACSLS 89
 - TLD 82
- robtest 84, 104
- robtest utility
 - Linux 55

S

- SAM utility
 - HP-UX 47
- SAN clients
 - about drivers for Linux 53
 - configuring drivers on AIX 14
 - configuring drivers on HP-UX 35
 - configuring drivers on Solaris 73
- schgr device driver
 - HP-UX 36
- Scripts
 - sg.install
 - Solaris 60
- scripts
 - sgscan 69, 76
- SCSI
 - pass through driver
 - Solaris 57
 - robotic control
 - HP-UX 33
 - Linux (2.6 kernel) 52
 - on Linux 51
 - on Solaris 69
 - SCSI persistent bindings 54
 - SCSI reservations
 - data integrity 9
 - disabling 9
 - disabling SPC-2 reserve in AIX 21
 - disabling SPC-2 reserve in HP-UX 47
 - disabling SPC-2 reserve on Solaris 72
 - sctl device file
 - creating for FCP (Itanium) 41
 - creating for FCP (PA-RISC) 39
 - creating for SCSI (PA-RISC) 37
 - sg driver
 - Linux 50
 - Solaris 57
 - uninstalling 75
 - sg.build command
 - Solaris 76
 - sg.conf file
 - example 63
 - sg.install script
 - Solaris 60, 76
 - sg.links file
 - example 64
 - shared ACS drives
 - configuring 95
 - smit command 16
 - Solaris
 - adapter card removal 57
 - binding Fibre Channel HBA drivers 59
 - command summary 75
 - configuring SAN clients 73
 - disabling MPxIO 60
 - disabling SPC-2 reserve on Solaris 72
 - examples of SCSI and FCP robotic control device files 69
 - introduction 56
 - locate-block 71
 - no rewind device files 71
 - preventing driver unloading 67
 - robotic controls 68–69
 - SCSI pass through driver 57

- Solaris (*continued*)
 - SCSI robotic controls 69
 - sg driver install or reconfigure 60
 - sg.install script 60
 - SPC-2 SCSI reserve 72
 - tape drive configuration 70
 - Berkeley-style close 71
 - using ACS 57
 - Solaris Multiplexed I/O (MPxIO)
 - disable 60
 - Sony S-AIT drives 22
 - SPC-2 SCSI reserve
 - disable in HP-UX 47
 - disabling in AIX 22
 - Solaris 72
 - SSO
 - configuring shared ACS drives 96
 - st driver
 - Linux 51
 - st tape driver
 - debug mode 51
 - st.conf file
 - example 63
 - Sun
 - acctest utility on UNIX 107
 - acctest utility on Windows 107
 - Automated Cartridge System
 - multiple ACS robots with multiple ACS hosts 109
 - multiple ACS robots with single ACS host 108
 - Automated Cartridge System (ACS)
 - configurations supported 108
 - Sun StorageTek ACSLS
 - firewall configuration 110
 - Sun StorEdge Network Foundation HBA driver
 - configuration 59
 - switch settings
 - Sony S-AIT
 - on AIX 22
- T**
- table-driven robotics 83
 - tape device drivers
 - Windows 78
 - tape drive access
 - persistent DSFs 29
 - Tape drive configuration
 - HP-UX
 - no rewind 28
 - tape drive configuration
 - on AIX
 - make device files 17
 - Solaris 70
 - tape drive device files
 - Linux (2.6 kernel) 52
 - requirements 27
 - tape drive device files in AIX
 - configuring 15
 - tape drive pass-through device files
 - creating 43
 - tape driver
 - choosing 15
 - tape drives
 - creating no rewind device files 17
 - legacy pass-through paths 34
 - nonstandard 73
 - tapes
 - removing from ACS robots 97
 - removing using NetBackup 98
 - removing using the ACSLS utility 98
 - text version of this guide 8
 - TLD robots 82
- U**
- uninstalling
 - the sg driver 75
 - UNIX
 - acctest utility 107
 - upgrading
 - NetBackup to use HP-UX persistent DSFs 31
 - using
 - guide 8
 - using the passthru driver
 - capabilities 58
- V**
- variable length block 16
 - variable-mode devices
 - on AIX 16
 - verifying
 - device configuration on Linux 52

W

Windows

- acstest utility 107

- attaching devices 78

- tape device drivers 78

- world wide node names (WWNN) 60

- world wide port names (WWPN) 59–60